TESIS KI142502

Deteksi Intrusi dengan Jumlah Jarak ke Centroid dan Sub-centroid

Kharisma Muchammad
5114201032

DOSEN PEMBIMBING
Tohari Ahmad, S.Kom., MIT., Ph.D.
NIP. 19750525200312 1 002

PROGRAM MAGISTER
BIDANG KEAHLIAN KOMPUTASI BERBASIS JARINGAN
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2016

(halaman ini sengaja dikosongkan)

TESIS KI142502

Detecting Intrusion using Distance to Centroid and Sub-Centroid

Kharisma Muchammad
5114201032

Advisor
Tohari Ahmad, S.Kom., MIT., Ph.D.
NIP. 19750525200312 1 002

PROGRAM MAGISTER
BIDANG KEAHLIAN KOMPUTASI BERBASIS JARINGAN
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2016

Tesis disusun untuk memenuhi salah satu syarat memperoleh gelar
Magister Komputer (M.Kom.)
di
Institut Teknologi Sepuluh Nopember Surabaya

oleh:
KHARISMA MUCHAMMAD
Nrp. 5114201032

Dengan judul :
Deteksi Intrusi dengan Jumlah Jarak ke Centroid dan Sub-centroid

Tanggal Ujian : 15-1-2016
Periode Wisuda : 2015 Gasal

Disetujui oleh:

Tohari Ahmad, S.Kom, MIT, Ph.D
NIP. 197505252003121002
(Pembimbing 1)

Dr.Eng. Radityo Anggoro, S.Kom, M.Sc
NIP. 198410162008121002
(Penguji 1)

Royyana Muslim I, S.Kom, M.Kom, Ph.D
NIP. 197708242006041001
(Penguji 2)

Henning Titi Ciptaningtyas, S. Kom, M. Kom
NIP. 198407082010122004
(Penguji 3)

Direktur Program Pasca Sarjana,

Prof.Ir.Djauhar Manfaat, M.Sc., Ph.D.
NIP. 196012021987011001

Deteksi Intrusi dengan Jumlah Jarak dari *Centroid* dan *Sub-centroid*.

Namamahasiswa	: Kharisma Muchammad
NRP	: 5114201032
Pembimbing I	: Tohari Ahmad, S.Kom., MIT., Ph.D.

ABSTRAK

Keamanan jaringan telah menjadi salah satu fokus dalam penelitian keamanan komputer. Salah satu cara untuk mendapatkan jaringan yang aman adalah dengan menggunakan sistem deteksi intrusi (*Intrusion Detection System*/IDS). Salah satu teknikdalam IDS yang banyak dipakai adalah *machine learning*. Dalam teknik *machine learning* tersebut, penggunaan fitur yang tepat dapat meningkatkan akurasi dan menurunkan kompleksitas komputasi yang diperlukan oleh program. Terdapat 2 pendekatan untuk mendapatkan fitur yang baik untuk proses *machine learning.* Pendekatan pertama adalah *feature selection*, dimana sejumlah fitur dipilih dari fitur yang sudah ada. Pendekatan kedua adalah *feature generation* dimana fitur dibangkitkan dengan mentransformasi fitur yang sudah ada.

Beberapa penelitian telah mengajukan metode untuk membangkitkan fitur pada IDS. Dari beberapa penelitian sebelumnya beberapa masalah yang dihadapi antara lain: metrik ekstraksi fitur yang mungkin gagal, penggunaan sumber daya (RAM, waktu) yang efisien. Permasalahan yang akan dibahas pada proposal ini adalah bagaimana meningkatkan kualitas fitur yang dihasilkan dan meningkatkan akurasi dari metode-metode yang telah diajukan sebelumnya.

Sejumlah penelitian telah dilakukan untuk menunjukkan penggunaan jarak dari *centroid* dapat dipakai untuk membangkitkan fitur yang dapat dipakai untuk klasifikasi data. Terdapat 2 sisi mengenai penggunaan tingkat homogenitas *cluster* dalam mencari *centroid* yang representatif, pendekatan pertama adalah mencari *centroid* tanpa pedulikan homogenitas *cluster* yang dihasilkan, sedangkan pendekatan lain adalah mencari *centroid* dari *cluster* yang sangat homogen. Penelitian ini mengajukan sebuah metode pembangkitan fitur satu dimensi menggunakan jumlah jarak dari *centroid* dan *subcentroid* dengan mempertimbangkan homogenitas *cluster* sebagai *soft-constraint*.

Berdasarkan ekeperimen yang dilakukan, metode yang diajukan dapat menurunkan waktu pengolahan sambil meningkatkan akurasi, *specificity*, dan sensitivitas. Pada dataset NSL-KDD 20% terjadi penurunan waktu pengolahan sebesar 10 menit dan pada dataset Kyoto2006 sebesar 7 jam. Akurasi metode ini lebih tinggi 4% daripada TANN pada dataset NSL-KDD20% dan 1% pada Kyoto2006. Sensitivitas yang dihasilkan lebih tinggi 2% daripada TANN pada NSL-KDD 20% dan 3% pada Kyoto2006. *Specificity* yang dihasilkan 6% lebih tinggi daripada TANN pada dataset NSL-KDD dan 2% pada dataset Kyoto2006.

Kata kunci: deteksi intrusi, keamanan jaringan, keamanan komputer

Detecting Intrusion Using Sum of Distance to Centroids and Sub-centroids

Name : Kharisma Muchammad
Student Identity Number :5114201032
Supervisor :Tohari Ahmad, S.Kom., MIT., Ph.D.

ABSTRACT

Network security has become a focus in computer security sesearch. One way to ensure network security is by using intrusion detection system (IDS). Machine learning approach is gaining attention in the field of intrusion detection. Because this approach depends on feature used to detect intrusion, selecting or generating good feature is a problem. A Good feature selection or generation can increase the accuracy of detection and decrease the complexity of the program.

Some study have proposed feature generation or selection method. There are some problems with the previous methods. Some of which is: Feature extraction might fail, relatively high resources consumption, In this study we are looking for a way to increase the quality of the feature generated and achieve better accuracy than previously proposed method.

Some researches have proposed using distance of the data to centroid to generate better feature(s) for classification. Those studies differ in the aspect of cluster homogeneity to extract centroid. Some studies disregard cluster homogenity while other creates homogeneous clusters.

In this study, we propose a feature generation method to generate one dimentional feature. This is generated by sum the distance of data to the centroids and subcentroids while taking account cluster homogeneity as soft constraint.

Based on the experiment, the proposed methods can decrease processing time and improve accudacy, sensitivity, and spesificity. In NSL-KDD 20% dataset, our proposed method shows 10 minute decrease and in Kyoto2006 7 hour decrease. The accuracy of our method is 4% higher than TANN on NSL-KDD and 1% higher on Kyoto2006. The sensitivity is 2% higher than TANN on NSL-KDD 20% and 3% higher on Kyoto2006. The spesificity is 6% higher than TANN on NSL-KDD 20% and 2% on Kyoto2006.

Key words: computer security, intrusion detection, network security

DAFTAR ISI

(halaman ini sengaja dikosongkan)

DAFTAR GAMBAR

DAFTAR TABEL

(halaman ini sengaja dikosongkan)

BAB 1
PENDAHULUAN

1.1 Latar Belakang

Pengamanan informasi pada zaman ini telah menjadi kebutuhan vital bagi berbagai pihak. Proses ini dilakukan dengan berbagai cara, antara lain enkripsi (Ahmad dkk, 2009), steganografi (Holil dan Ahmad, 2015), atau autentifikasi biometrik (Ahmad dan Hu, 2010). Keamanan jaringan menghadapi beberapa masalah seperti akses yang tak terotorisasi dan penggunaan sumber daya yang tidak diijinkan. Salah satu solusi untuk menangani masalah tersebut adalah dengan menggunakan sistem deteksi intrusi /*intrusion detection system* (IDS). IDS sendiri dapat dibagi menjadi 2 yaitu IDS berbasis *signature* dan berbasis deteksi anomali.

IDS berbasis *signature* menggunakan basis data aktifitas yang diaggap sebagai serangan. Jika ada aktifitas yang sesuai dengan *signature* maka aktifitas tersebut akan dianggap sebagai serangan dan sistem akan mengaktifkan alarm.

IDS berbasis deteksi anomali menggunakan suatu model normal untuk memberikan gambaran aktifitas apa saja yang dianggap normal. Sistem memeriksa perbedaan antara aktifitas yang diperiksa dengan model normal untuk menentukan apakah alarm harus diaktifkan atau tidak.

Kelebihan metode deteksi anomali dibanding metode *signature* adalah kemampuan metode ini dalam menghadapi serangan yang tidak terdapat dalam basis data (Garcia-Teodoro dkk, 2009). Salah satu kekurangan dari metode anomali dibanding *signature* adalah sumber daya yang diperlukan untuk mendeteksi serangan relatif lebih besar daripada metode yang menggunakan *signature* karena proses yang dilakukan relatif lebih komplek (Garcia-Teodoro dkk, 2009). Kelemahan lain dari metode deteksi anomali adalah tingkat *false positive* yang lebih tinggi dari pada metode *signature* (Garcia-Teodoro dkk, 2009).

Garcia-Teodoro dkk, (2009) membagi teknik deteksi anomali menjadi tiga kelompok, metode berbasis statistik, berbasis pengetahuan dan *machine learning.*

Metode *machine learning* sendiri dibagi menjadi 6 kelas antara lain *bayessian network, markov model, neural network, fuzzy logic, genetic algorithm* dan deteksi *outlier* dengan *clustering*. Sommer dan Paxon (2010) memaparkan beberapa masalah yang ditemukan dalam pendekatan *machine learning*. Masalah masalah tersebut antara lain:

a. Karakter dari deteksi *outlier*: dalam metode deteksi *outlier* biasanya model deteksi dibentuk dari hanya menggunakan data aktifitas normal dan menganggap data diluar data *training* tadi anomali. Menurut Witten dan Frank (2005), pendekatan ini disebut ausmsi dunia tertutup (*closed world assumption*) dimana asumsi ini tidak banyak berguna di dunia nyata.
b. Biaya akibat galat: berbeda dengan domain lain seperti sistem rekomendasi produk, deteksi spam, maupun pengenal tulisan tangan, tingkat *false positive* dan *false negative* pada IDS dituntut seminimal mungkin. Aktifitas positif menuntut keluarnya waktu dan tenaga yang untuk memeriksa aktivitas tersebut. Jika banyak terjadi *false positive* banyak pula waktu dan tenaga yang dikeluarkan menjadi sia-sia. Mengabaikan hasil positif karena terlalu banyak *false positive* juga bukan tindakan yang baik karena ada kemungkinan adanya *true positive* yang perlu ditindaklanjuti sesegera mungkin.

Beberapa masalah yang dikemukakan oleh Sommer di atas telah ditangani oleh beberapa peneliti. Lin dkk (2015) dan Tsai dkk (2010) mampu menghasilkan tingkat deteksi yang tinggi sambil menjaga tingkat *false positive* rendah menggunakan teknik *clustering* dan pembangkitan fitur (*feature generation*). Namun metode mereka memiliki kelemahan yaitu sumber daya yang diperlukan untuk proses *learning* relative tinggi karena pada tahap akhir, klasifikasi data *testing* harus dibandingkan dengan seluruh data *training*.

Sejumlah penelitian seperti (Han dan Karypis, 2000), (Lin dkk, 2015) dan (Guo dkk, 2014) berpendapat jarak data ke *centroid* dari suatu dataset dapat dipakai sebagai fitur untuk membedakan data dalam dataset tersebut. Namun, pendekatan ini menggunakan seluruh dataset yang telah tertransformasi sebagai data *training.* Penggunaan jarak ke *subcentroid* diharapkan dapat memberikan

tambahan kemampuan untuk membedakan data dalam dataset yang lebih kecil, dalam satu *cluster*, sehingga jumlah data yang dipakai sebagai data *training* lebih kecil tanpa menurunkan akurasi dan pada akhirnya menurunkan *running time*.

Penggunaan homogenitas *cluster* dalam proses *clustering* diharapkan dapat memberikan *centroid* yang lebih representatif terhadap data. Hal ini dilakukan karena sejumlah metode sebelumnya seperti (Lin dkk, 2015), (Tsai dkk, 2009) dan (Guo dkk, 2014) tidak mempertimbangkan homogenitas *cluster* saat mendapatkan *centroid*. Beberapa penelitian seperti (Han dan Karypis, 2000) mendapatkan *centroid* dengan merata-rata seluruh data yang selabel sehingga *cluster* yang dihasilkan sangat homogen. Perbedaan (Han dan Karypis, 2000) dengan penelitian ini adalah dari dataset yang digunakan dan rumus jarak yang digunakan dimana (Han dan Karypis, 2000) menggunakan *cosine distance* dan penelitian ini menggunakan *euclidean distance*.

1.2 Perumusan Masalah

Rumusan masalah yang diangkat padapenelitianini adalah sebagai berikut:

a. Bagaimana pembangkitan fitur (*feature generation*) dilakukan ?
b. Faktor apa yang mempengaruhi homogenitas *cluster* ?
c. Apakah ada pengaruh penggunaan jarak ke *subcentroid* terhadap akurasi maupun *running time* ?
d. Apakah ada pengaruh homogenitas *cluster* terhadap performa deteksi ?

1.3 Batasan Masalah

Dataset yang dipakai adalah NSL-KDD 20% dan sebagian dari dataset Kyoto2006++.

1.4 Tujuan Penelitian

Tujuan penelitian ini adalah mendapatkan model pembangkitan fitur untuk IDS yang lebih baik daripada metode sebelumnya dari sisi akurasi maupun *running time*.

1.5 Manfaat Penelitian

Manfaat pada penelitian ini adalah memberikan metode baru dalam membangkitkan fitur untuk mendeteksi serangan.

1.6 Kontribusi Penelitian

Kontribusi yang diberikan oleh penelitian ini adalah pembaruan model dalam pembangkitan fitur untuk deteksi intrusi dengan mempertimbangkan tingkat homogenitas *cluster* dan jarak ke *subcentroid* dan tanpa memerlukan pengetahuan mengenai jumlah *cluster* yang diperlukan.

BAB 2
DASAR TEORI

2.1 Sistem Deteksi Intrusi / *Intrusion Detection System* (IDS)

Penelitian mengenai IDS telah dimulai sejak tahun 1987 oleh Denning (1987). Dalam penelitiannya, Denning mengemukakan idenya mengenai model pengamanan dari sistem yang sudah ada menggunakan sub-sistem IDS dengan asumsi aktifitas yang mengeksploitasi kelemahan sistem secara statistik berbeda dengan aktifitas normal.

2.2 KDD99 Dataset

Dataset KDD99 adalah dataset yang dipakai dalam kompetisi Kdnuggets pada tahun 1999. Dataset ini dikembangkan dari TCP *dump* dataset DARPA 1998. Dataset DARPA 98 dibentuk dari jaringan tertutup dan menggunakan *network traffic generator* untuk mensimulasikan aktifitas normal dan serangan. Aktifitas serangan maupun normal yang terlalu komplek untuk dapat diotomasi masih harus dilakukan secara manual. Dataset Darpa 1998 berukuran 4 gigabyte dalamkondisi terkompresi dalam format zip hasil dari tcpdump trafik jaringan selama 7 minggu yang jika diolah menghasilkan sekitar 5.000.000 *record* koneksi. KDD99 berisi sekitar 4.900.000 *record* koneksi dimana tiap koneksi terdiri dari 41 fitur dan dilabeli serangan atau normal.

Kelas serangan sendiri terdiri dari 4 kelas serangan, antara lain.

a. *Denial of Service* (DoS) : kelas serangan ini membebani sumber daya komputer baik berupa *processor*, ram, maupun *bandwidth* jaringan sehingga komputer korban kesulitan menghadapi koneksi yang asli.
b. *User to Root* (U2R) : kelas serangan ini secara umum berusaha untuk mendapatkan akses *root*/admin pada komputer korban. Searangan ini memanfaatkan celah keamanan yang dapat dimanfaatkan jika penyerang telah mendapatkan akses sebagai pengguna yang sah entah itu dengan cara resmi, *social engineering, sniffing* (menyadap koneksi pengguna resmi), maupun serangan *dictionary* dengan mencoba kata sandi yang umum dipakai.

c. *Remote to Local* (R2L) : Kelas serangan ini secara umum berupaya untuk mendapatkan akses ke komputer korban sebagai pengguna resmi.
d. *Probing* : kelas serangan ini secara umum bertujuan untuk mencari informasi mengenai jaringan komputer yang akan diserang. Salah satu contoh adalah serangan ping untuk memeriksa apakah komputer dengan ip tertentu ada atau tidak. Contoh lain adalah serangan *port scan* untuk melihat port mana yang terbuka dari suatu komputer.

Dalam dataset *training* probabilitas distribusi serangan berbeda dengan probabilitas kemunculan pada data *training.* Beberapa pakar melihat serangan baru secara umum adalah variasi dari serangan yang sudah dikenali sebelumnya. Sehingga dalam penggunaan IDS *signature* serangan lama masih dapat dipakai untuk mendeteksi variasi serangan baru.

Meski dataset ini populer di kalangan peneliti IDS, beberapa peneliti mempermasalahkan penggunaan dataset ini. McHugh (2000) mengutarakan kritiknya atas metodologi yang dipakai dalam membangun dataset DARPA 1998. Kritik tersebut antara lain:

a. Pada saat itu tidak ada produk komersil untuk validasi hasil.
b. Statistik untuk membangun *background traffic* tidak dipublikasikan.
c. Distribusi serangan tidak dipastikan terdistribusi secara realistis.

Problem lain juga diutarakan oleh (Tavallaee dkk, 2009). Permasalahan itu antara lain:

a. Banyak data yang redundan
b. Dataset relatif mudah

Dalam penelitiannya, Tavallaee dkk (2009) mengajukan dataset yang lebih sulit diolah (NSL-KDD) agar performa IDS dapat dibandingkan dengan lebih baik.

2.3 Dataset NSL-KDD

Dataset ini diajukan oleh Tavallaee dkk, (2009) sebagai solusi atas problem dalam membentuk dataset KDD99. Dalam penelitiannya mereka melihat ada 2 masalah yang terdapat dalam dataset KDD99 tersebut. Masalah tersebut antara lain banyaknya data yang redundan dan data yang terlalu mudah.

Data yang redundan dapat mengakibatkan proses belajar mengalami bias kepada data yang redundansinya tinggi. Hal ini lebih jauh lagi mengakibatkan proses belajar kesulitan mendeteksi data yang tidak sering muncul seperti kelas U2r, atau R2l.

Penelitian IDS dengan KDD secara umum menggunakan teknik *machine learning* mencari pola pada dataset yang dapat dipakai untuk membedakan aktivitas serangan dan normal. Untuk mencapai hal ini dataset dipecah menjadi 2 kelompok, kelompok *training* dan kelompok *tesing* dimana teknik *machine learning* dilatih dengan dataset *training* dan divalidasi dengan data *tesing.*

Dalam penelitiannya Tavallaee dkk (2009), menguji 7 teknik *machine learning* yang terdapat dalam Weka. Tiap teknik dilatih 3 kali dengan 3 dataset *training* yang berbeda sehingga dihasilkan 21 model deteksi. 21 model deteksi ini kemudian dipakai untuk menguji seberapa sulit tiap *record* diklasifikasi dengan nilai antara 0-21 dimana 0 adalah yang paling sulit untuk dideteksi dan 21 yang paling mudan untuk dideteksi. Hasil ekemerimen mereka menunjukkan 93 % dari dataset KDD dapat dideteksi oleh seluruh model deteksi. Untuk mengatasi hal ini mereka melakukan proporsi ulang terhadap dataset KDD99 sehingga dihaslkan dataset NSL-KDD.

2.4 Kyoto2006

Dalam penelitiannya Song dkk, (2011) berpendapat dataset KDD99 kurang dapat merefleksikan kondisi jaringan di dunia nyata. Mereka menggunakan pendekatan yang berbeda dengan KDD99 dalam mendapatkan data koneksi. Dalam penelitian tersebut, mereka menggunakan sejumlah *honeypot* untuk mendapatkan koneksi jaringan nyata. *Honeypot* adalah komputer yang disediakan untuk menjebak penyerang dan merekam aktifitas penyerang.

Untuk proses pelabelan koneksi, mereka menggunakan 3 perangkat lunak yaitu Symantec Network Security 7160, ClamAV dan Ashula. Untuk mendapatkan *traffic* normal, meraka memasang 2 *server* asli pada jaringan *honeypot*. *Server* yang dipasang adalah *mailing server* dan DNS *server*. Kedua *server* tersebut membuka beberapa layanan seperti ssh, http, dan https untuk memudahkan manajemen. Mereka melakukan pengumpulan data selama 3 tahun mulai dari tahun 2006 hingga akhir 2009.

2.5 K-*means* clustering

K-*means* merupakan salah satu algoritma *clustering* yang sering dipakai untuk menklasifikasikan data dari berbagai disiplin ilmu. Metode ini bertujuan mempartisi data menjadi K partisi dimana jarak antara data ke pusat partisinya/*centroid* minimal. Karena problem ini bersifat NP-*hard* dipakai pendekatan *heuristic* untuk mencari titik pusat partisi. Langkah-langkah pada pendekatan *heuristic* tersebut antara lain:

a. Cari K *centroid* partisi secara acak, ulangi langkah b dan c hingga *centroid* tidak bergeser
b. Partisi data berdasarkan jarak ke *centroid* terdekat.
c. Cari *centroid* baru dengan mencari rata-rata dari tiap partisi.

Salah satu problem dari metode ini adalah perlunya parameter K untuk mengetahui jumlah partisi yang diperlukan. Pada beberapa kasus nilai K ini tidak diketahui.

2.6 K-nearest neighbor

K-nearest neighbor (K-nn) adalah salah satu metode klasifikasi data yang paling sederhana. Metode ini bekerja dengan mencari k tetangga terdekat dari suatu data yang tidak dikenali pada suatu dataset *training*. Data yang tidak dikenali tadi kemudian diklasifikasi berdasarkan label dari k tetangga yang dominan. Penentuan nilai K yang rendah membuat metode klasifikasi rentan terhadap *outlier* dan berakibat misklasifikasi.

2.7 Triangle Area and Nearest Neighbor (TANN)

TANN merupakan metode deteksi intrusi yang diajukan oleh Tsai dan Lin (2010).Alur metode ini dapat dilihat pada Gambar 2.1. Metode ini terdiri dari beberapa langkah. Langkah-langkah itu ialah:

a. Ekstraksi *centroid* dengan *clustering.*

   Langkah ini dilakukan dengan menggunakan metode *clustering* K-*means* dan nilai k=5.

b. Membentuk segitiga.

   Segitiga yang dibentuk terdiri dari data yang akan ditransformasi dan 2 *centroid*. Jumlah segitiga yang dihasilkan adalah $C_{5,2} = 10$. Total luas seluruh segitiga tadi dijumlahkan untuk mendapatkan fitur baru yang

dipakai untuk proses deteksi. Proses pembentukan segitiga dapat dilihat pada Gambar 2.2.

c. Deteksi.

Data yang diperoleh dari langkah b dipecah menjadi 10 bagian untuk dilakukan 10-*fold cross validation.* Langkah ini menggunakan K-*Nearest Neighbor*.

Metode ini memiliki beberapa permasalahan. Pertama, nilai k untuk K-*means* yang dipakai diambil dari jumlah kelas yang terdapat pada dataset KDD99. Jika dataset diganti kemungkinan besar nilai k tidak relevan. Permasalahan lain yang mungkin terjadi adalah jika tiga titik yang dipakai untuk membuat segitiga tadi membentuk garis lurus atau data yang akan ditransformasikan berada di *centroid* maka tidak ada segitiga yang dapat dihasilkan.

2.8 Bisecting K-*means*

Bisecting k-*means* diajukan oleh Steinbach dkk (2000) untuk klasifikasi dokumen. Algoritma ini bekerja sebagai berikut:

a. Semua data dimasukkan dalam satu *cluster*
b. Pilih *cluster* yang akan dipecah
c. Cari 2 sub-*cluster* dengan algoritma K-*means* biasa
d. Ulangi langkah b dan c sampai jumlah *cluster* yang diinginkan tercapai

2.9 Gini *Impurity Index*

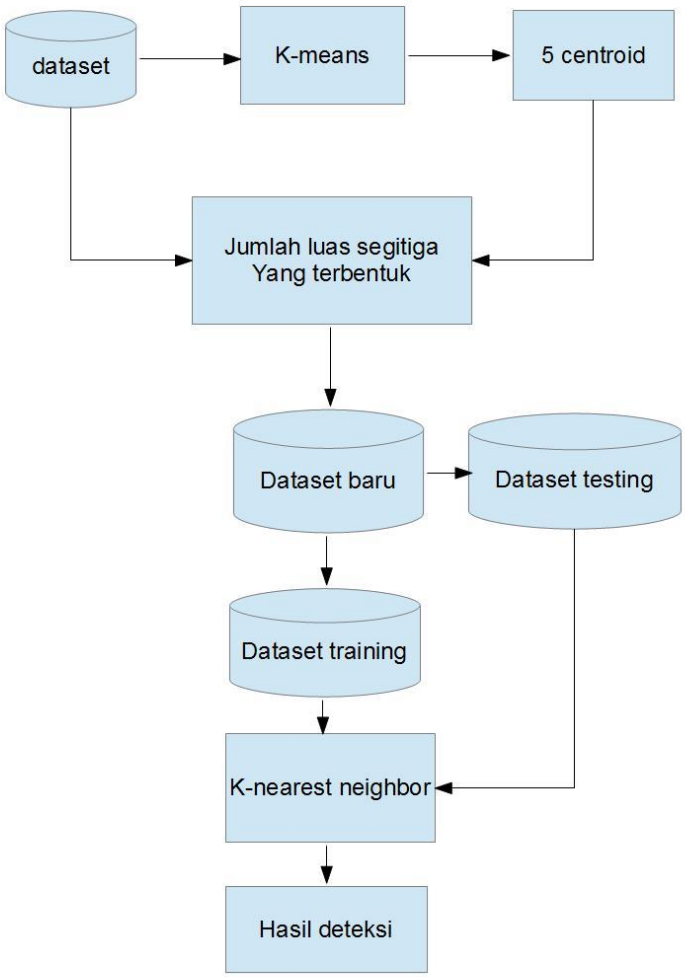
Gini *impurity* index adalah index untuk mengukur ketidakmurnian suatu himpunan. *Index* ini didapatkan dengan persamaan (2.1) dimana I_G adalah nilai gini *impurity index*, m adalah jumlah label dalam himpunan tersebut dan f_i adalah frekuensi/proporsi dari label ke-i

$$I_G = 1 - \sum_{i=1}^{m} f_i^2 \quad \textbf{(2.1)}$$

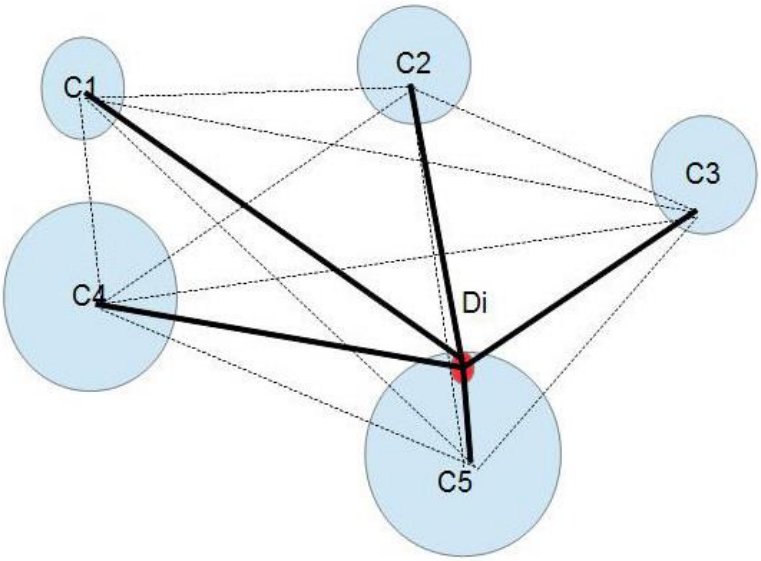
$$H = -\sum_{i=0}^{m} f_i log_2 f_i \quad \textbf{(2.2)}$$

2.10 Shannon *Entropy*

Entropy dapat didefinisikan sebagai ketidakpastian dari informasi yang terkandung dalam suatu pesan. *Entropy* dari suatu pesan didapatkan dengan persamaan (2.2) dimana H adalah nilai *entopy* himpunan, m adalah jumlah elemen yang ada dalam suatu pesan dan f_i adalah frekuensi kemunculan elemen ke-i dalam pesan.



Gambar 2.1 Alur TANN



Gambar 2.2. Proses membentuk segitiga dari data Di dan 2 dari 5 *centroid*

Halaman ini sengaja dikosongkan

BAB 3
METODE PENELITIAN

Bab ini membahas langkah penelitian. Langkah penelitian terdiri dari 6 tahap yaitu studi literatur, desain sistem, implementasi sistem, pengujian sistem, analisis hasil, dan penyusunan buku.

3.1 Tahapan Penelitian

Metodologi yang dilakukan dalam penelitian ini memiliki beberapa tahapan yang dilakukan agar dapat mendapatkan hasil yang sesuai dengan tujuan penelitian. Terdapat enam tahap pada penelitian ini yang dapat dilihat di Gambar 3.1. Agar dapat lebih mudah dipahami dan dimengerti, tiap-tiap proses yang ada di dalam tahapan penelitian ini akan dijelaskan secara lebih rinci pada sub-bagian berikut.

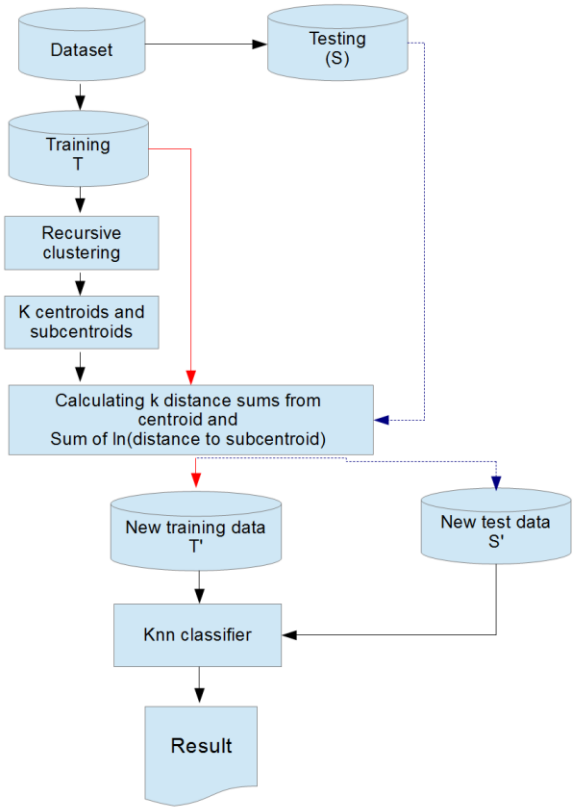
3.1.1. Studi Literatur

Pada langkah ini dilakukan pencarian dan pengumpulan literatur yang berhubungan dengan penelitian ini. Sumber literatur yang dicari berupa literatur primer seperti makalah ilmiah dan proseding konferensi.

Gambar 3.1 Enam Tahapan Penelitian

3.1.2. Desain Sistem

Pada langkah ini dilakukan desain sistem untuk memperbaiki metode yang sudah ada. Alur sistem yang diajukan dalam penelitian ini dapat dilihat pada Gambar 3.2. Metode yang diusulkan pada penelitian ini memiliki 2 langkah yaitu pembangkitan fitur dan klasifikasi.



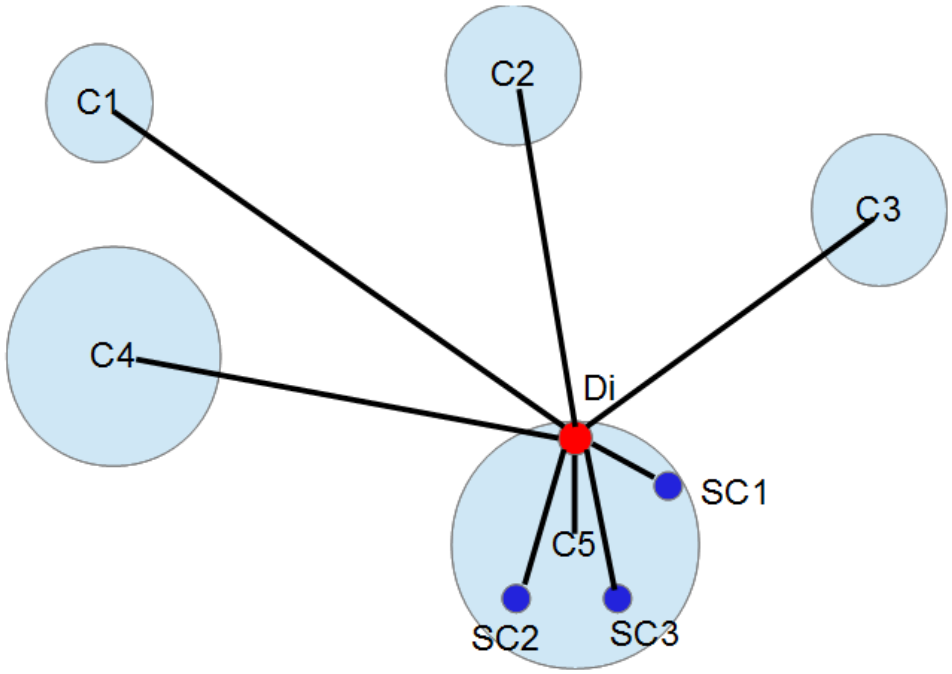
Gambar 3.2. Alur Sistem yang Diajukan

3.1.2.1.Pembangkitan fitur / *Feature Generation*

Pada langkah ini dilakukan proses *clustering* pada dataset *training* T dengan *recursive clustering*. Langkah *recursive clustering* sendiri dilakukan dengan men-*cluster* data menjadi 2 *cluster* dengan K-*means*. Jika *cluster* yang dihasilkan memiliki gini *impurity index* atau *entropy* melebihi ambang batas yang dimasukkan pengguna, U, maka *cluster* tersebut dipecah lagi menjadi 2 *cluster*. Hal ini dilakukan sampai setiap *cluster* memiliki gini *impurity index* atau *entropy* dibawah ambang batas atau jumlah data dalam *cluster* tersebut terlalu kecil untuk di-*cluster*. Gini *impurity* index dapat dilihat di subbab 2.5 dan *entropy* dapat

dilihat di subbab 2.6 dimana f_i adalah frekuensi label ke-i. Proses berikutnya yang dilakukan adalah mencari *subcentroid* di tiap *cluster*. *Subcentroid* didapatkan dengan melakukan k-*means clustering* pada tiap *cluster* dengan nilai k sama dengan nilai parameter pengguna, O, jika jumlah data dalam *cluster* tersebut lebih dari O.

Setelah *cluster*, *centroid* dan *subcentroid* didapatkan, proses berikutnya memasukkan tiap $D_i \in T$ ke *centroid* yang paling dekat. Gambar 3.3 menunjukkan jarak apa saja yang diperlukan untuk membangkitkan fitur. Pada gambar tersebut didapatkan 5 *centroid* dan data D_i masuk ke *cluster* 5. Untuk mendapatkan D_i' jarak yang perlu dijumlah adalah jarak D_i ke tiap *centroid cluster* dan logaritma natural dari D_i ke tiap *subcentroid* (SC1-SC3) dari tiap *subcentroid* dari *cluster* 5.



Gambar 3.3. Jarak yang Diperlukan untuk Membangkitkan Fitur

Proses pembangkitan fitur untuk data *testing* S dilakukan dengan memasukkan data *testing* ke *cluster* dengan jarak ke *centroid*-nya paling dekat. Langkah berikutnya sama dengan pembangkitan fitur pada data *training* yaitu dengan mencari jarak dari data ke tiap *centroid cluster* dan logaritma natural dari jarak data ke sub *centroid*.

Keluaran dari langkah ini adalah T' dan S' yaitu data *training* dan *testing* yang telah ditransformasikan. Fitur yang dibangkitkan pada langkah ini berupa fitur 1 dimensi.

3.1.2.2. Klasifikasi

Fase klasifikasi dilakukan dengan k-*nearest neighbor* namun tidak semua data training T' dipakai. Untuk menklasifikasi data *testing* hanya dilakukan K-*Nearest Neigbor* terhadap data *training* yang masih satu *cluster* dengan data itu. Nilai K yang dipakai dalam percobaan ini adalah 3 saja. Hal ini dilakukan untuk menyederhanakan eksperimen yang dilakukan.

3.1.3. Pengujian Sistem

Untuk menguji metode yang diajukan, dilakukan implementasi dengan bahasa pemograman Python dan SciKit. Metode yang dipakai untuk *benchmark* adalah TANN. Eksperimen dilakukan dengan beberapa nilai parameter. Teknik validasi yang dipakai adalah 10-*fold cross validation* dimana dataset dipecah menjadi 10 bagian. Terdapat 9 bagian untuk *training* dan 1 bagian untuk *testing*. Dari validasi tadi akan dihasilkan 10 skenario uji untuk tiap parameter pada diap metode.

3.1.3.1.Eksperimen Metode yang diusulkan

Nilai parameter dari metode yang diusulkan dapat dilihat pada Tabel 3.1. Dalam tiap eksperimen, kriteria uji homogenitas yang dipakai adalah *entropy*, gini *impurity* index, dan gabungan keduanya. Persamaan untuk gabungan dapat dilihat pada persamaan (3.1) dimana I_C adalah Indeks gabungan, I_G adalah indeks gini *cluster* tersebut yang dapat dilihat pada persamaan (2.1) dan H adalah nilai *entropy cluster* tersebut yang dapat dilihat pada persamaan (2.2). Persamaan tersebut didesain agar nilai yang dihasilkan berkisar antara 0 sampai 1 dengan memberikan bobot yang sebanding antara Gini *impurity index* dan *entropy*.

Nilai parameter 0.3 dipakai karena nilai ini sudah mencapai lebih dari setengah nilai gini *impurity index* (nilai gini *impurity index* hanya berkisar antara 0-0,5). Asumsi *subcentroid* yang dipakai tidak lebih dari 6 karena 5 adalah ukuran cluster paling besar yang hampir pasti homogen dan dibiarkan. Pada cluster

beranggotakan 5 nilai gini paling kecil yang diatas 0 adalah 0,32. Karena nilai ini masih berada diatas threshold maka clsuter tersebut masih harus dipecah lagi.

Tabel 3.1 Nilai Parameter yang Diuji

No	O (*impurity* index)	U (Asumsi jumlah sub-*cluster*)
1	0.1	4
2	0.1	5
3	0.1	6
4	0.2	4
5	0.2	5
6	0.2	6
7	0.3	4
8	0.3	5
9	0.3	6

$$I_C = \left(\frac{2\ x\ I_G + H}{2}\right) \quad \textbf{(3.1)}$$

Nilai asumsi jumlah sub-*cluster* tidak kurang dari 4 karena nilai asumsi ini harus dapat mengakomodasi *cluster* yang anggotanya banyak dan heterogen. *Cluster* yang besar dapat diwakili hanya dengan 2 atau 3 sub-*cluster*, namun secara umum akan lebih baik jika diwakili lebih dari 3 sub-*cluster*.

3.1.3.2.Eksperimen TANN

Eksperimen dengan TANN akan dilakukandengannilai K yang digunakan untuk klasifier k-nn pada tiap eksperimen adalah 3,5,7,9,11,13, dan 15 atau sampai terjadi penurunan nilai akurasi, sensitivitas dan *specificity*. Nilai 1 tidak dipakai untuk mengurangi dampak *outlier* pada fase klasifikasi.

3.1.3.3. Eksperimen Metode yang Diusulkan dengan K-*means* Menggantikan Recursive Clustering

Eksperimen ini dilakukan untuk mengetahui pengaruh homogenitas *cluster* yang terbentuk terhadap akurasi deteksi serangan. Nilai K pada k-*means* adalah 5 dan rata-rata jumlah *cluster* yang didapatkan pada eksperimen 3.1.3.1 yang menghasilkan nilai akurasi paling tinggi.

3.1.3.4. Eksperimen Metode yang Diusulkan tanpa Menggunakan Fungsi Logaritma

Eksperimen in dilakukan untuk mengetahui pengaruh penggunaan log pada jarak antara data dengan *subcentroid.* Nilai parameter dan *impurity index* yang dipakai adalah parameter dan *impurity* index yang terbaik yang didapatkan pada eksperimen 3.1.3.1.

3.1.4. Dataset

Dataset yang dipakai dalam eksperimen ini adalah NSL-KDD 20% dan Kyoto2006. Atribut yang dihilangkan dari NSL-KDD adalah protocol_type, service, dan flag. Hal ini dilakukan karena ketiga atribut itu bertipe simbolik. Dataset Kyoto2006 yang dipakai adalah tanggal 20 juli 2009 dan atribut yang dibuang adalah flag, start_time dan duration. Atribut flag dibuang karena atribut ini bersifat simbolik, start_time tidak dipakai karena tidak terlihat relevan, dan duration dihilangkan karena varian dari kolom ini terlalu besar. Data yang redundan pada dataset Kyoto2006 kemudian dibuang.

3.1.5. Analisis Hasil

Pada langkah ini dilakukan analisa terhadap hasil yang didapat dari eksperimen. Metrik yang diambil dari eksperimen dapat dibagi menjadi 2 kelompok. Kelompok pertama adalah akurasi, sensitivitas, dan *specificity*. Ketiga metrik diatas menunjukkan tingkat kebenaran dari keluaran yang dihasilkan. Persamaan untuk mendapatkan akurasi, sensitivitas dan *specificity* dapat dilihat pada persamaan (3.2), (3.3) dan (3.4) dimana TP adalah serangan yang berhasil dideteksi sebagai serangan, TN adalah aktivitas normal yang berhasil dideteksi sebagai aktivitas normal, FP adalah aktifitas normal yang dideteksi sebagai serangan dan FN adalah serangan yang gagal dideteksi sebagai serangan. Akurasi menggambarkan aktivitas yang terklasifikasi dengan benar. Sensitivitas menggambarkan rasio serangan yang berhasil dideteksi dibandingkan serangan yang seharusnya terdeteksi. *Specificity* menggambarkan rasio aktifitas normal yang berhasil dideteksi dibandingkan aktifitas normal yang seharusnya terdeteksi. Sensitivitas dan *specificity* dipakai untuk mendapatkan gambaran lebih baik mengenai ketepatan klasifikasi data. Metode klasifikasi bisa saja mendapatkan akurasi dan sensitivitas tinggi namun *specificity* rendah dengan kondisi mayoritas

dataset *testing* berisi serangan dan selalu melabeli aktifitas sebagai serangan. Hal yang sebaliknya juga bisa terjadi, metode deteksi memberikan nilai akurasi dan *specificity* tinggi namun sensitivitas rendah dengan kondisi mayoritas dataset *testing* berisi data normal dan selalu melabeli aktifitas sebagai aktifitas normal. Metode deteksi yang baik akan memberikan hasil yang tinggi pada ketiga nilai kriteria tersebut.

Tipe serangan yang gagal dideteksi dan berhasil dideteksi juga akan diperiksa. Kelompok kedua adalah analisis kompleksitas dari metode yang diuji, parameter yang diuji adalah *running time* algoritma.

$$\boldsymbol{Akurasi} = \frac{\boldsymbol{TP} + \boldsymbol{TN}}{\boldsymbol{TP} + \boldsymbol{TN} + \boldsymbol{FP} + \boldsymbol{FN}} \quad \textbf{(3.2)}$$

$$\boldsymbol{Sensitivitas} = \frac{\boldsymbol{TP}}{\boldsymbol{TP} + \boldsymbol{FN}} \quad \textbf{(3.3)}$$

$$\boldsymbol{Spesificity} = \frac{\boldsymbol{TN}}{\boldsymbol{TN} + \boldsymbol{FP}} \quad \textbf{(3.4)}$$

3.1.6. Penyusunan Buku

Tahap terakhir merupakan penyusunan laporan yang memuat dokumentasi mengenai pembuatan serta hasil dari implementasi dari sistem yang telah dibuat.

Halaman ini sengaja dikosongkan

BAB 4
HASIL DAN PEMBAHASAN

4.1 Hasil TANN pada NSL-KDD

Nilai akurasi, *specificity*, sensitivitas dan waktu pengolahan TANN pada dataset NSL-KDD dapat dilihat pada Tabel 4.1. Dapat dilihat pada tabel tersebut nilai akurasi dan sensitivitas meningkat seiring dengan nilai K, namun nilai *specificity* menurun seiring peningkatan K. Hal ini disebabkan kecenderungan data uji dikelilingi data serangan. Jika memang data uji memang data serangan, sensitivitas akan menigkat, namun jika data uji merupakan data normal maka *specificity* akan menurun. Tidak terlihat ada korelasi yang jelas antara waktu pengolahan dan nilai K yang dipakai. Antara nilai K=3 ke K=5 terjadi penurunan, namun dari K=5 ke K=7 terjadi peningkatan waktu pengolahan, namun secara umum waktu pengolahan berkisar diantara 0,5 jam (30 menit).

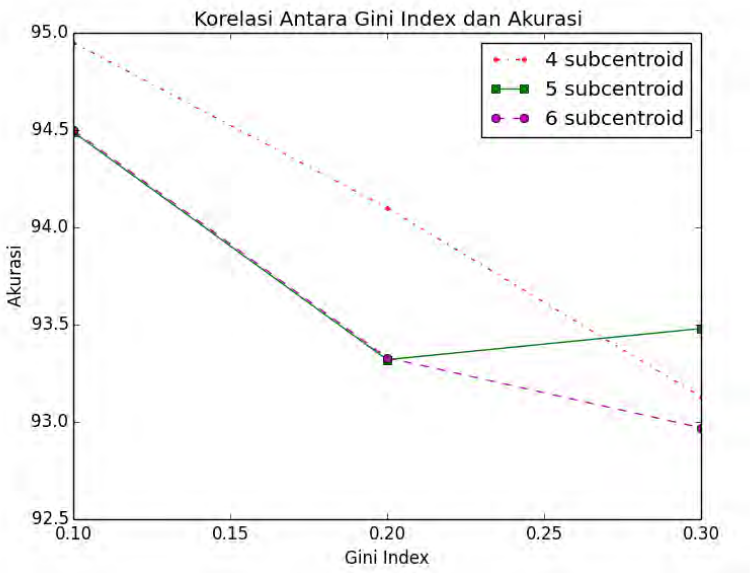
Tabel 4.1 Hasil Eksperimen TANN pada NSL-KDD

Nilai K	Akurasi (%)	Sensitivitas (%)	*Specificity* (%)	Waktu Pengolahan (jam)
3	93,08	94,85	91,53	0,47
5	93,36	95,86	91,17	0,39
7	93,51	96,32	91,05	0,56
9	93,56	96,48	91,02	0,42
11	93,60	96,60	90,99	0,50
13	93,61	96,66	90,95	0,53
15	93,54	96,62	90,86	0,53

4.2 Hasil Metode yang Diajukan dengan Gini *Impurity Index* pada NSL-KDD

Nilai akurasi, sensitivitas, *specificity*, rata-rata jumlah *cluster*, dan waktu pengolahan metode yang diajukan dengan gini *impurity* index dapat dilihat pada Tabel 4.2 untuk akurasi, Tabel 4.3 untuk sensitivitas, Tabel 4.4 untuk *specificity*, Tabel 4.5 untuk jumlah *cluster*, dan Tabel 4.6 untuk waktu pengolahan. Korelasi nilai gini *impurity* index dan akurasi, sensitivitas, dan *specificity* dapat dilihat pada Gambar 4.1 untuk akurasi, Gambar 4.2 untuk sensitivitas dan Gambar 4.3 untuk *specificity*. Berdasarkan tabel tersebut dapat dilihat nilai akurasi, sensitivitas, dan *specificity* yang paling tinggi diperoleh dengan nilai parameter

impurity index 0,1 dan asumsi *subcentroid* 4. Secara umum pertumbuhan nilai akurasi dan sensitivitas menurun seiring dengan peningkatan nilai *impurity* index dan asumsi *subcentroid*. Hal yang sama tidak dapat dikatakan pada nilai *specificity*. Nilai *specificity* memang menurun jika *impurity* index ditingkatkan tanpa menigkatkan nilai asumsi *subcentroid*, namun pada parameter gini *impurity* index 0,2 dan 0,3 terjadi peningkatan seiring nilai asumsi *subcentroid* meskipun nilai spesifitas ini tidak sebaik nilai *specificity* eksperimen dengan parameter gini *impurity* 0,1 dan asumsi *subcentroid* 5. Hal ini menunjukkan peningkatan asumsi jumlah *subcentroid* dapat sedikit mengkompensasi penurunan *specificity* yang terjadi jika nilai gini *impurity* index ditingkatkan.

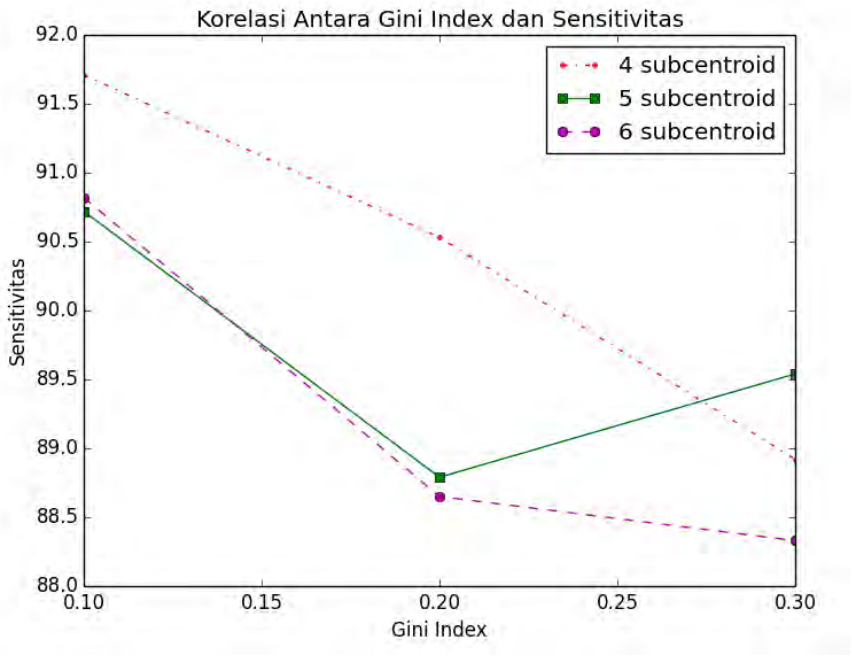


Gambar 4.1 Korelasi antara Nilai Gini *Index* dan Akurasi pada NSL-KDD

Dibandingkan dengan TANN, terjadi peningkatan akurasi dan *specificity*, tapi terjadi penurunan pada sensitivitas. Peningkatan akurasi terjadi dari 93.54% menjadi 94,95%. Peningkatan *specificity* terjadi dari 91,53% menjadi 97,77%. Penurunan sensitivitas terjadi dari 96,62% ke 91,71%.

Berdasarkan Tabel 4.5 dan 4.6 terdapat indikasi korelasi antara waktu pengolahan dan jumlah *cluster* terhadap nilai *impurity* index, makin rendah *impurity* index makin banyak jumlah *cluster* yang dihasilkan dan makin lama waktu yang diperlukan untuk mengolah data. Hal ini terjadi karena algoritma

divisive hierarchial clustering yang dipakai memerlukan waktu lebih untuk mencapai *impurity* index yang ditargetkan. Dibandingkan dengan TANN, metode yang diajukan dengan gini *impurity* index memerlukan waktu lebih sedikit. Metode TANN paling sedikit memakan waktu 0,39 jam dan metode yang diajukan paling lama memerlukan waktu 0,31 jam.



Gambar 4.2 Korelasi antara Nilai Gini *Index* dan Sensitivitas pada NSL-KDD

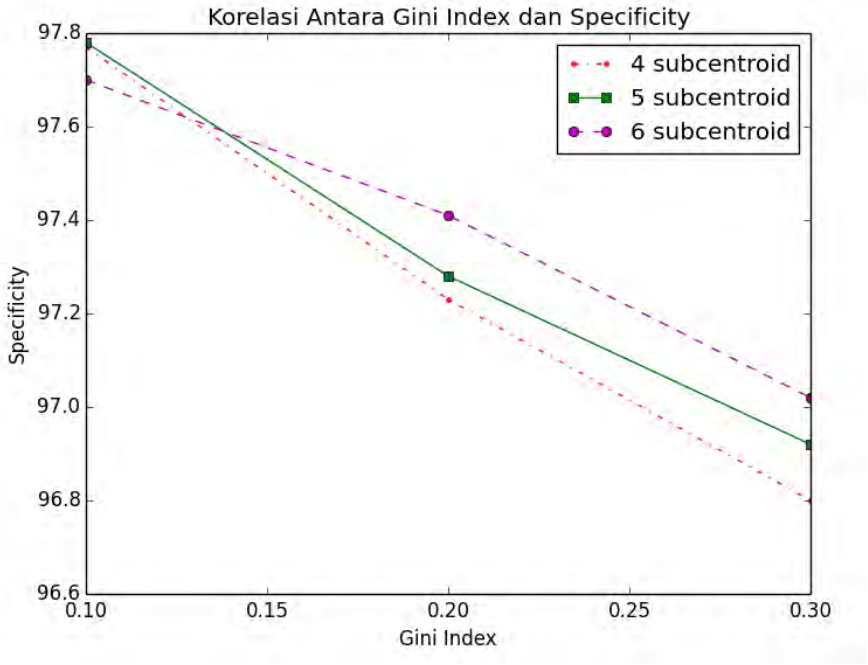
Tabel 4.2 Nilai Akurasi Metode yang Diajukan dengan Gini *Index* pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai Gini *Impurity*		
	0,1	0,2	0,3
4	94,95	94,10	93,13
5	94,49	93,32	93,48
6	94,50	93,32	92,97

Tabel 4.3 Nilai Sensitivitas Metode yang Diajukan dengan Gini *Impurity Index* pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai Gini *Impurity*		
	0,1	0,2	0,3
4	91,71	90,53	88,92
5	90,72	88,79	89,54

6	90,82	88,65	88,33



Gambar 4.3 Korelasi antara Nilai Gini *Index* dan *Specificity* pada NSL-KDD

Tabel 4.4 Nilai *Specificity* Metode yang Diajukan dengan Gini *Impurity Index* pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai Gini *Impurity*		
	0,1	0,2	0,3
4	97,77	97,23	96,80
5	97,78	97,28	96,92
6	97,70	97,41	97,02

Tabel 4.5 Jumlah Cluster Rata-rata Metode yang Diajukan dengan Gini *Impurity Index* pada NSL-KDD

Asumsi *subcentroid*	Nilai Gini *Impurity*		
	0,1	0,2	0,3
4	541,7	354	218,4
5	546,7	351,6	217,5
6	539	347,8	217,2

Tabel 4.6 Waktu Pengolahan Data Metode yang Diajukan dengan Gini *Impurity Index* pada NSL-KDD dalam Jam

Asumsi *subcentroid*	Nilai Gini *Impurity*		
	0,1	0,2	0,3
4	0,315	0,225	0,161
5	0,268	0,205	0,15
6	0,311	0,273	0,161

4.3 Hasil Metode yang Diajukan dengan *Entropy* pada NSL-KDD

Nilai akurasi, *specificity*, sensitivitas, jumlah *cluster* yang dihasilkan dan waktu pengolahan metode yang diajukan dengan *entropy* dapat dilihat pada Tabel Tabel 4.7 untuk akurasi, Tabel 4.8 untuk sensitivitas, Tabel 4.9 untuk *specificity*, Tabel 4.10 untuk jumlah *cluster*, dan Tabel 4.11 untuk waktu pengolahan. Korelasi nilai gini *impurity* index dan akurasi, sensitivitas, dan *specificity* dapat dilihat pada Gambar 4.4 untuk akurasi, Gambar 4.5 untuk sensitivitas dan Gambar 4.6 untuk *specificity*. Berdasarkan tabel tersebut dapat dilihat secara umum peningkatan nilai *entropy* berdampak pada penurunan nilai ketiga kriteria uji. Fenomena yang sama terjadi pada pengaruh perubahan nilai asumsi *subcentroid*. Pada Tabel 4.7 nilai akurasi, sensitivitas, dan *specificity* tertinggi didapat dengan nilai asumsi *subcentroid* 4. Hal ini menunjukkan kebanyakan *cluster* yang terbentuk membutuhkan 4 sub-*cluster* untuk memberikan hasil terbaik.

Nilai ketiga kriteria penilaian ini menunjukkan peningkatan dibandingkan TANN dan metode ini dengan gini *impurity* index. Salah satu faktor yang menyebabkan hal ini adalah jumlah *cluster* yang dihasilkan oleh *entropy* lebih banyak dibandingkan dengan gini index. Nilai akurasi terbesar pada gini *impurity* index dicapai dengan 541 *cluster* sedangkan pada *entropy* dicapai dengan 716 *cluster*.

Tabel 4.7 Nilai Akurasi Metode yang Diajukan dengan *Entropy* pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	98,28	95,28	94,71

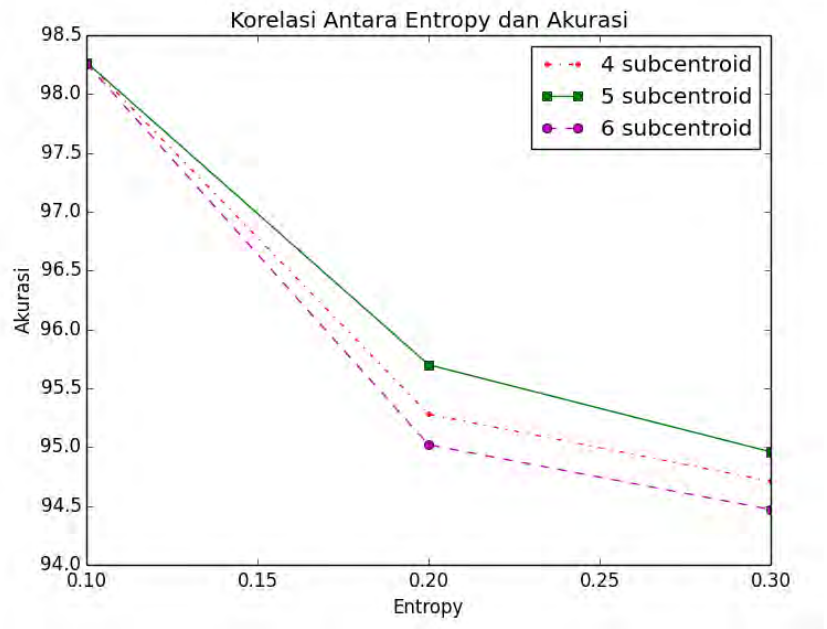
5	98,27	95,70	94,96
6	94,89	94,36	94,19

Tabel 4.8 Nilai Sensitivitas Metode yang Diajukan dengan *Entropy* pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	98,88	92,64	91,20
5	98,65	93,40	91,73
6	91,73	90,87	90,93

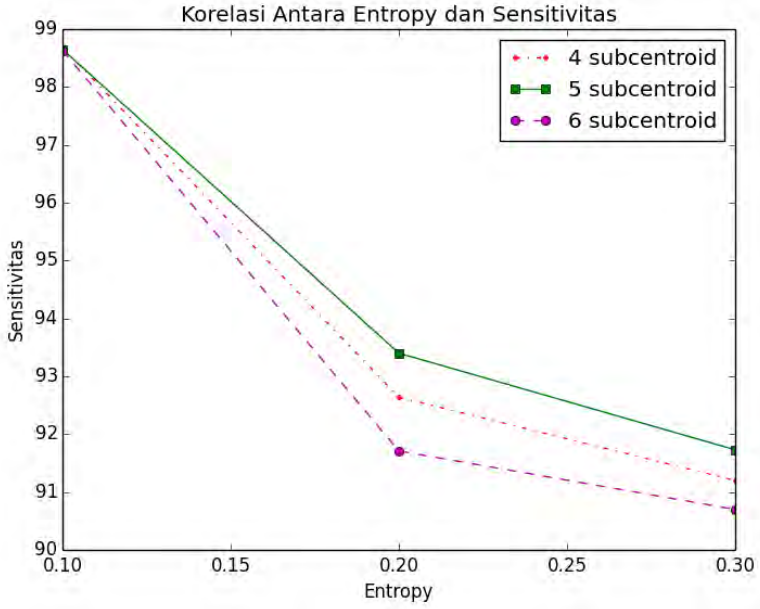
Tabel 4.9 Nilai *Specificity* Metode yang Diajukan dengan *Entropy* pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	97,93	97,58	97,77
5	97,94	97,70	97,77
6	97,65	97,40	97,03

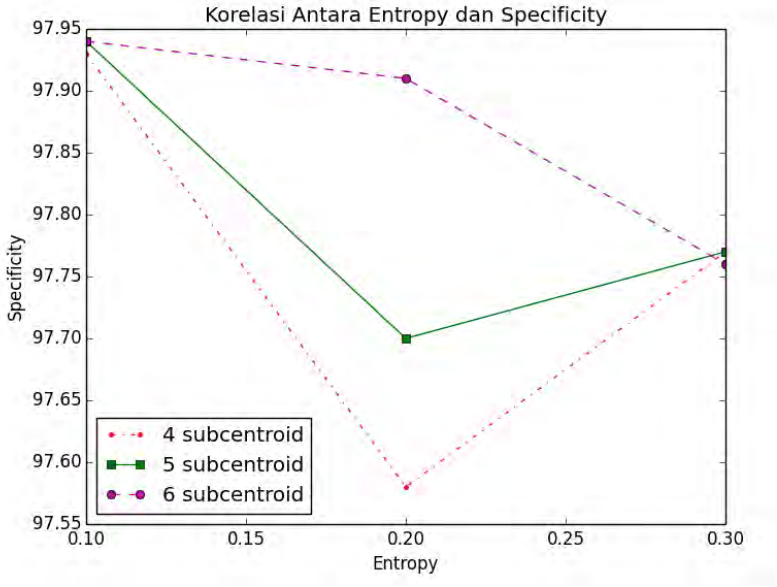


Gambar 4.4 Korelasi antara Nilai *Entropy* dan Akurasi pada NSL-KDD

Dari tabel 4.10 dan 4.11 terlihat indikasi korelasi antara nilai *entropy* dengan jumlah *cluster*. Makin rendah nilai *entropy* makin tinggi jumlah *cluster* yang dihasilkan. Tidak terlihat adanya korelasi antara waktu pengolahan dan nilai *entropy*.



Gambar 4.5 Korelasi antara Nilai *Entropy* dan Sensitivitas pada NSL-KDD



Gambar 4.6 Korelasi antara Nilai *Entropy* dan *Specificity* pada NSL-KDD

Tabel 4.10 Jumlah Cluster Rata-rata dari Metode yang Diajukan dengan *Entropy* pada NSL-KDD

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	716,4	593,4	540,6
5	715,9	596	541,3
6	716	596,7	542,4

Tabel 4.11 Waktu Pengolahan Data dengan Metode yang Diajukan dengan entropy pada NSL-KDD dalam Jam

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	0,343	0,353	0,336
5	0,348	0,348	0,341
6	0,386	0,415	0,356

4.4 Hasil Metode yang Diajukan dengan Indeks Gabungan pada NSL-KDD

Nilai akurasi, *specificity*, sensitivitas, jumlah *cluster* dan lama pengolahan dari metode yang diajukan dengan indeks gabungan dapat dilihat pada Tabel 4.12 untuk akurasi, Tabel 4.13 untuk sensitivitas, Tabel 4.14 untuk *specificity*, Tabel 4.15 untuk jumlah *cluster*, dan Tabel 4.16 untuk waktu pengolahan. Korelasi nilai gini *impurity* index dan akurasi, sensitivitas, dan *specificity* dapat dilihat pada Gambar 4.7 untuk akurasi, Gambar 4.8 untuk sensitivitas dan Gambar 4.9 untuk *specificity*. Secara umum hasil ini hampir mirip dengan *entropy*. Nilai akurasi terbesar sama-sama dicapai pada index 0,1 dan asumsi *subcentroid* 4, dengan nilai akurasi dari *entropy* sebesar 98,28% dan gabungan sebesar 98,31%. Perbedaan terjadi pada sensitivitas dan *specificity*, nilai sensitivitas pada *entropy* dicapai pada asumsi *subcentroid* 4 dan pada gabungan dicapai pada asumsi *subcentroid* 5 namun nilai sensitivitas yang dicapai sama-sama 98,68%. Nilai spesifitas yang tertinggi dicapai oeh *entropy* dan indeks gabungan tidak berbeda jauh, dengan nilai 97,94% untuk *entropy* dan 98,08% untuk indeks gabungan. Hal ini disebabkan perilaku fungsi ini hampir sama, sebagaimana dapat dilihat pada

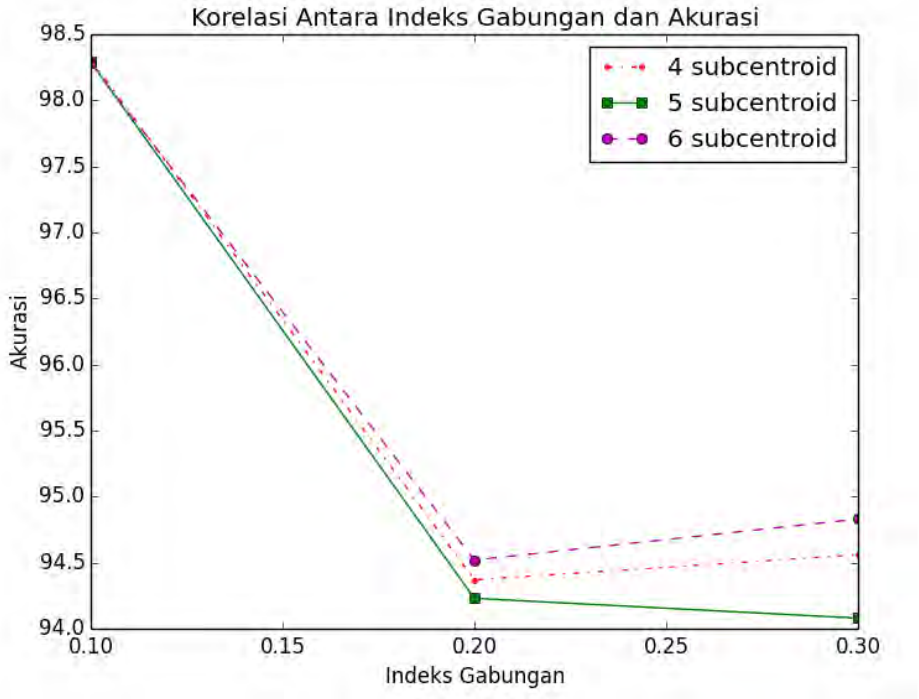
Gambar 4.11. Perbedaan antara gabungan dan *entropy* terjadi pada komposisi label antara 0 dan 0.5 dan pada nilai komposisi serangan pada 0 dan 0.5 sama.

Tabel 4.12 Nilai Akurasi Metode yang Diajukan dengan Indeks Gabungan pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	98,31	94,37	94,56
5	98,29	94,23	94,08
6	98,28	94,52	94,83

Tabel 4.13 Nilai Sensitivitas Metode yang Diajukan dengan Indeks Gabungan pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	98,56	90,44	91,07
5	98,68	90,02	89,94
6	98,62	90,84	91,55



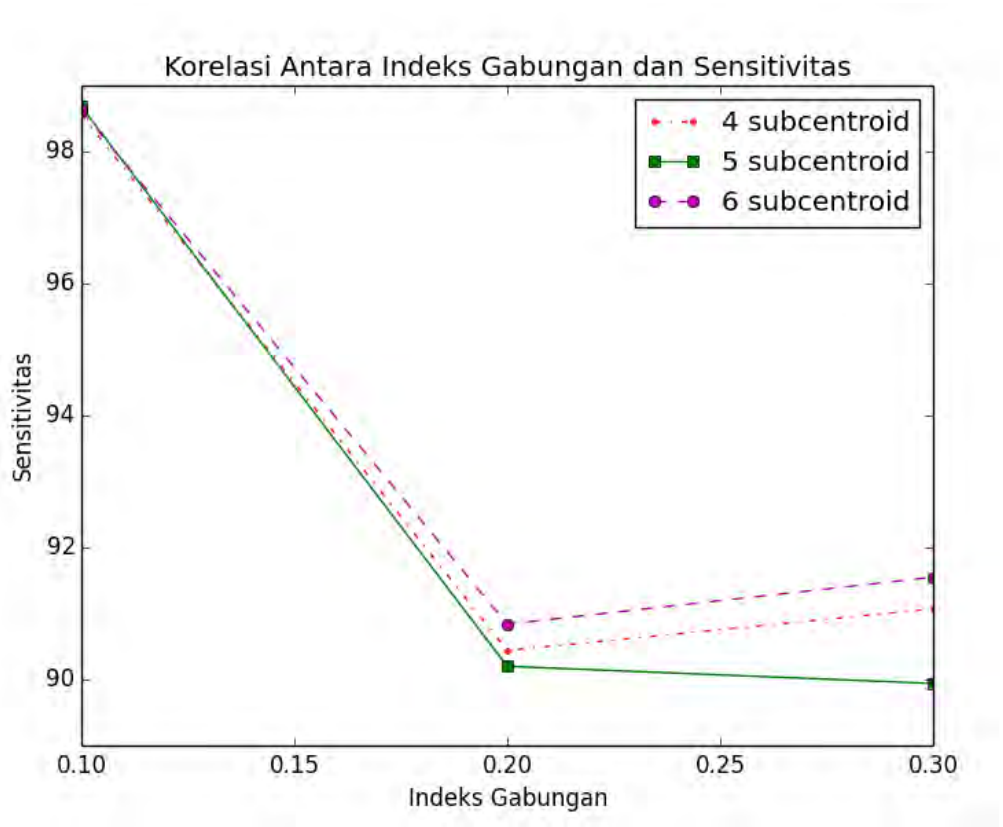
Gambar 4.7 Korelasi antara Nilai Indeks Gabungan dan Akurasi pada NSL-KDD

Tabel 4.14 Nilai *Specificity* Metode yang Diajukan dengan Indeks Gabungan pada NSL-KDD dalam Persen

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	98,08	97,79	97,62
5	97,95	97,74	97,69
6	97,97	97,73	97,70

Tabel 4.15 Jumlah Cluster Rata-rata dari Metode yang Diajukan dengan Indeks Gabungan pada NSL-KDD

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	**705.1**	569	507.7
5	704.2	656.2	512
6	703.2	565.8	510.8



Gambar 4.8 Korelasi antara Nilai Indeks Gabungan dan Sensitivitas pada NSL-KDD

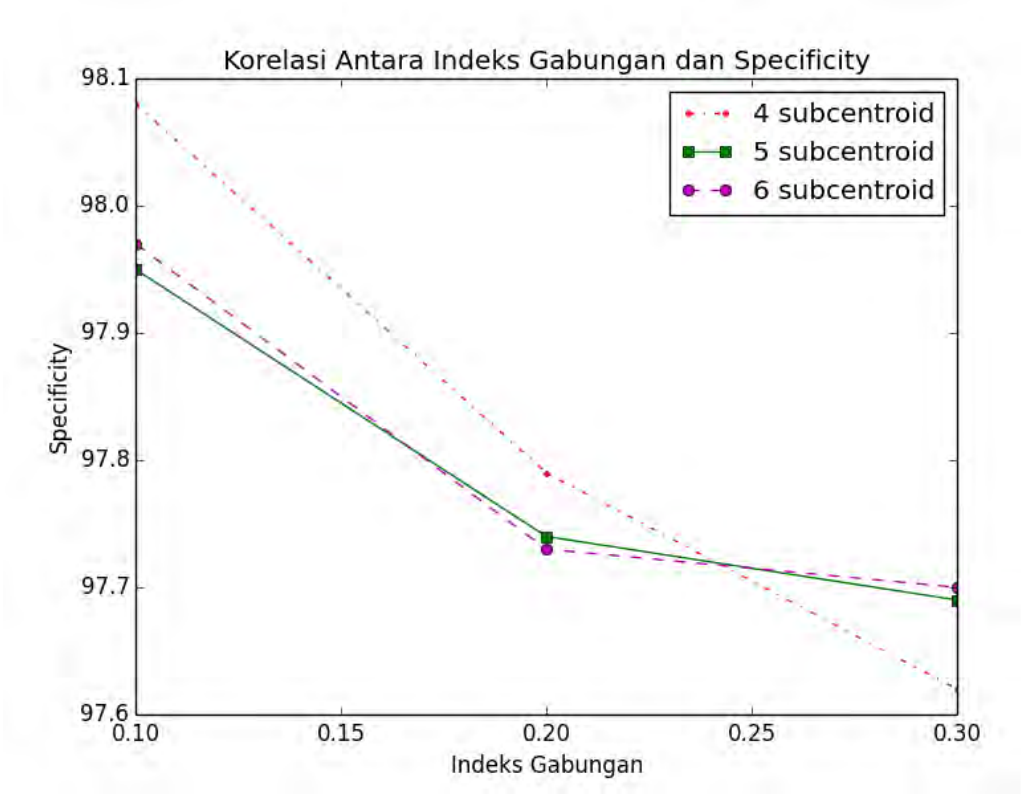
Dari ketiga indeks yang dipakai (Gini, *entropy*, dan gabungan), nilai akurasi terbaik didapatkan dengan penggunaan nilai indeks gabungan 0.1 dan dengan jumlah rata-rata *cluster* yang dihasilkan 705. Nilai ini akan dipakai untuk eksperimen penggunaan K-*means* secara langsung.

4.5 Hasil Metode yang Diajukan dengan Menggunakan K-*means* Pada NSL-KDD

Nilai akurasi, spesifitas, sensitivitas, jumlah *cluster* dan lama pengolahan dari metode yang diajukan dengan K-*means* biasa dapat dilihat pada Tabel 4.17 dan perbandingan dengan TANN, dan metode terbaik pada *bisecting* K-*means* dapat dilihat pada Gambar 4.10. Dari tabel tersebut dapat dilihat pengaruh algoritma *bisecting clustering* terhadap kriteria penilaian. K-*means* dengan 705 *cluster* memiliki akurasi lebih rendah daripada *bisecting* K-*means* dengan indeks gabungan 0,1. Pada indeks gabungan 0,1 akurasi yang dicapai sebesar 98,31%, lebih tinggi daripada akurasi dengan K-*means* biasa dengan jumlah *cluster* yang sama sebesar 97,44%. Sensitivitas dan *specificity* juga mengalami penurunan dibandingkan dengan indeks gabungan 0,1. Penurunan yang terjadi sebesar 1,12% pada sensitivitas dan 0,74% pada *specificity*. Meski terjadi penurunan pada ketiga kriteria di atas, waktu pengolahan juga mengalami penurunan dibandingkan dengan *bisecting* K-*means* dengan index gabungan 0,1. Penurunan terjadi dari 0,361 jam menjadi 0,27. Penurunan ini mungkin disebabkan pada bisecting K-*means clustering* dilakukan beberapa kali dan pada metode ini *clustering* hanya dilakukan sekali.

Tabel 4.16 Waktu Pengolahan Metode yang Diajukan dengan Indeks Gabungan pada NSL-KDD dalam Jam

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	0,361	0,348	0,331
5	0,360	0,366	0,331
6	0,293	0,278	0,261

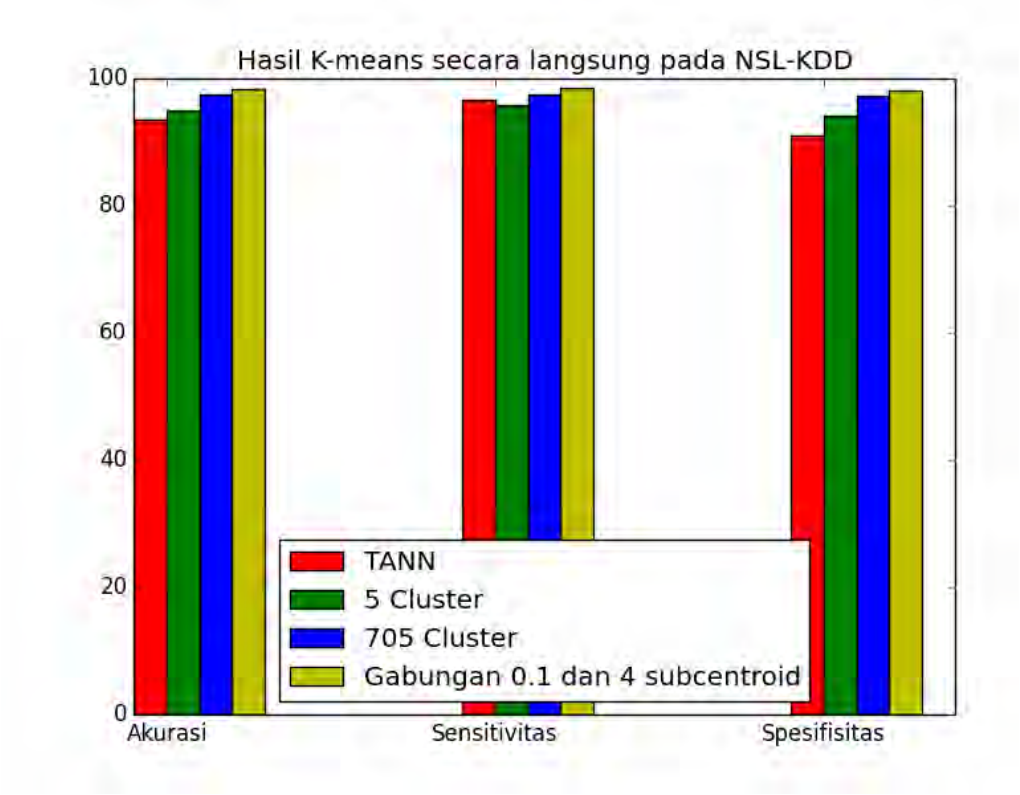


Gambar 4.9 Korelasi antara Nilai Indeks Gabungan dan *Specificity* pada NSL-KDD

Tabel 4.17 Hasil Metode yang Diajukan dengan K-*means* Menggantikan Bisecting K-*means* pada NSL-KDD

Jumlah Cluster	Akurasi (%)	Sensitivitas (%)	*Specificity* (%)	Waktu pengolahan (jam)
5	94,99	95,89	94,20	0,51
705	97,44	97,54	97,33	0,27

Pada eksperimen dengan K-*means* biasa dengan 5 *cluster*, nilai akurasi dan *specificity* masih lebih baik daripada TANN dengan 5 *cluster* namun sensitivitas metode ini lebih rendah daripada TANN. Nilai akurasi metode ini lebih tinggi 1,90%, *specificity* lebih tinggi 3,14% dan nilai sensitivitas mengalami penurunan sebesar 0,43%. Waktu pengolahan pada K-*means* dengan 5 *cluster* tidak terlihat perbedaan yang signifikan.

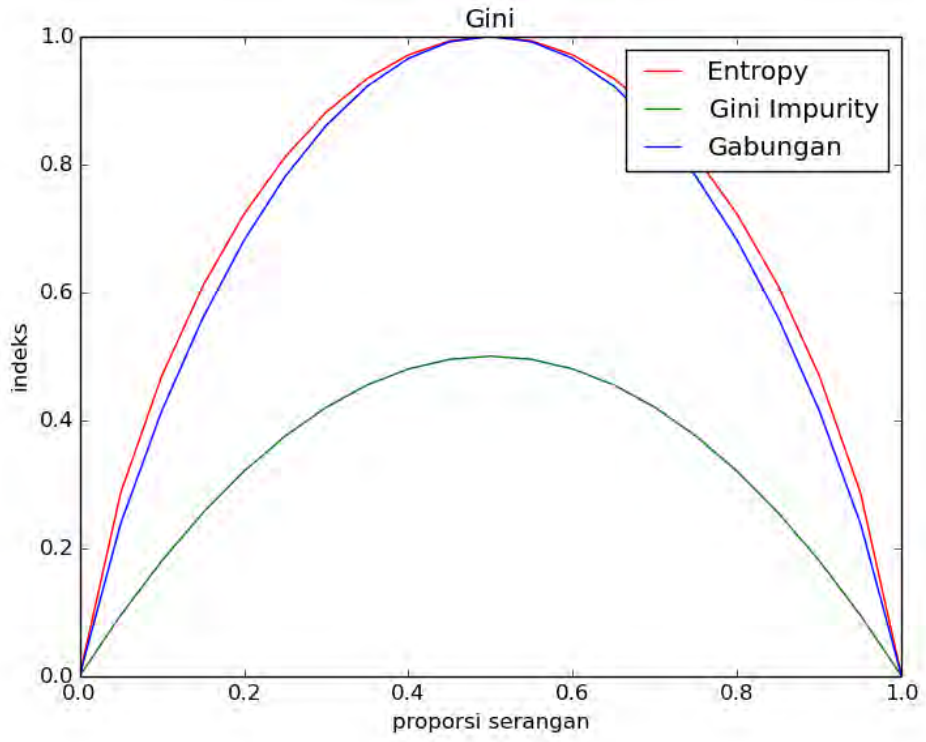


Gambar 4.10 Perbandingan Hasil K-*means* secara Langsung

4.6 Hasil Metode yang Diajukan Tanpa Logaritma pada NSL-KDD

Hasil eksperimen metode yang diajukan tanpa logaritma antara data dengan *subcentroid* dapat dilihat pada Tabel 4.18. Indeks *impurity* yang dipakai adalah gabungan dengan nilai 0.1 dan asumsi *subcentroid* 4. Dapat dilihat pada Tabel 4.18 nilai akurasi dari metode yang dihasilkan tidak berbeda jauh dari hasil terbaik dengan penggunaan logaritma. Hasil akurasi terbaik penggunaan jarak logaritma sebesar 98,31 %. Hal ini dikarenakan ukuran rata-rata *cluster* yang dihasilkan kecil sehingga jarak antar data pada *cluster* relatif rapat. Hal ini menyebabkan penggunaan fungsi logaritma tidak berdampak jauh. Nilai sensitivitas eksperimen ini lebih tinggi daripada eksperimen dengan menggunakan nilai indeks *impurity* dan asumsi *subcentroid* yang sama meski tidak berbeda jauh. Pada penggunaan jarak log dengan indeks gabungan 0.1 dan asumsi *subcentroid* 4 nilai sensitivitas yang dicapai sebesar 98,56% dan pada eksperimen ini 98,63% namun hal ini diikuti dengan penurunan *specificity* dari 98,08% menjadi 97,97%. Hal ini menunjukkan penggunaan jarak langsung pada dataset NSL-KDD dapat

mengelompokkan serangan pada sebagian *feature space*, namun membaurkan sebagian serangan pada *feature space* yang lain.



Gambar 4.11 Perilaku Fungsi yang Dipakai

Tabel 4.18 Hasil metode yang Diajukan Tanpa Logaritma pada NSL-KDD

Indeks gabungan	Akurasi (%)	Sensitivitas (%)	*Specificity* (%)	Waktu pengolahan (jam)
0.1	98,28	98,63	97,97	0.41

4.7 Hasil TANN pada Kyoto2006

Nilai akurasi, *specificity*, sensitivitas dan waktu pengolahan TANN pada dataset Kyoto2006 dapat dilihat pada Tabel 4.19. Nilai optimum K didapatkan dengan nilai K=5 dengan akurasi 96,80%, sensitivitas 95,92% dan 97,62%. Waktu pengolahan mengalami peningkatan dibandingkan NSL-KDD 20% karena jumlah data yang diolah 4 kali lipat lebih banyak daripada NSL-KDD 20%.

Dibandingkan NSL-KDD 20% waktu pengolahan meningkat sebesar 16 kali lipat hal ini menunjukkan waktu pengolahan metode ini tumbuh secara kuadratik.

4.8 Hasil Metode yang Diajukan dengan Gini *Impurity Index* pada Kyoto2006

Nilai akurasi, *specificity*, sensitivitas, rata-rata jumlah *cluster*, dan waktu pengolahan metode yang diajukan dengan gini *impurity* index dapat dilihat pada Tabel 4.20 untuk akurasi, Tabel 4.21 untuk sensitivitas, Tabel 4.22 untuk *specificity*, Tabel 4.23 untuk jumlah *cluster*, dan Tabel 4.24 untuk waktu pengolahan. Korelasi nilai gini *impurity* index dan akurasi, sensitivitas, dan *specificity* dapat dilihat pada Gambar 4.12 untuk akurasi, Gambar 4.13 untuk sensitivitas dan Gambar 4.14 untuk *specificity*. Dari tabel tersebut terlihat perbedaan karakteristik dataset Kyoto2006 dan NSL-KDD. Jumlah asumsi *subcentroid* yang diperlukan untuk mencapai nilai akurasi, sensitivitas, dan *specificity* terbaik. Dalam dataset NSL-KDD nilai terbaik didapatkan dengan jumlah asumsi *subcentroid* 4 sedangkan pada dataset Kyoto2006 nilai akurasi terbaik didapatkan dengan jumlah asumsi *subcentroid* 6. Jumlah *cluster* yang dihasilkan oleh metode ini dengan dataset NSL-KDD lebih banyak daripada Kyoto2006 meskipun jumlah data Kyoto2006 lebih banyak daripada NSL-KDD.

Tabel 4.19 Hasil TANN pada Dataset Kyoto2006

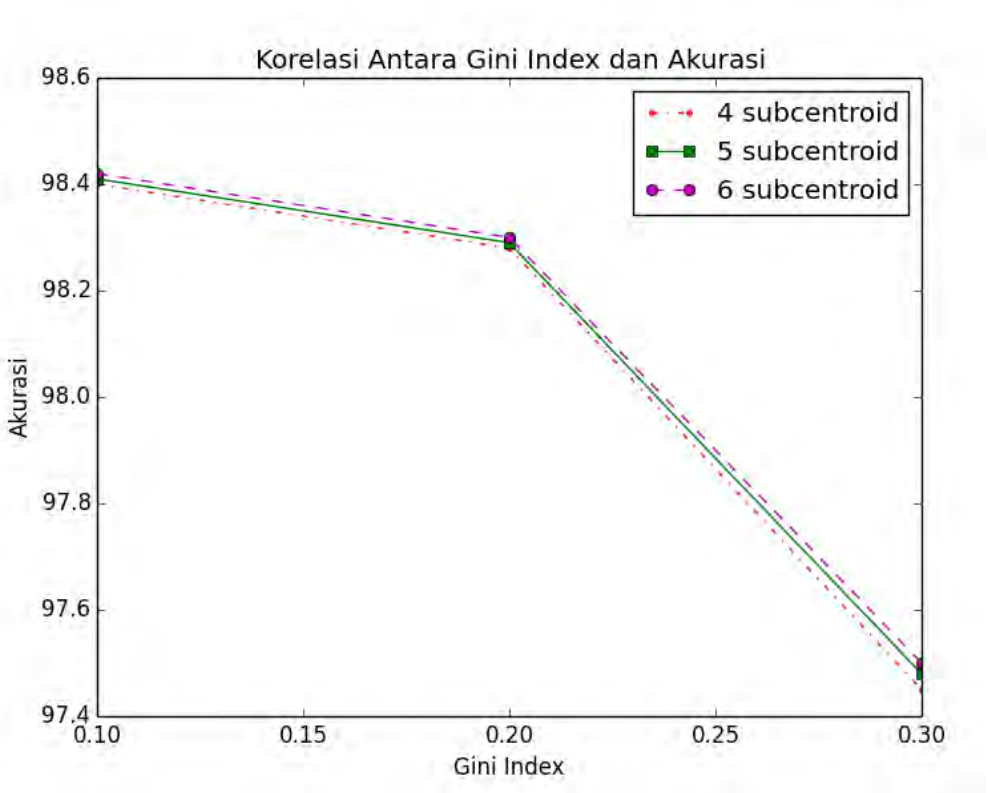
Nilai K	Akurasi (%)	Sensitivitas (%)	*Specificity* (%)	Waktu Pengolahan (jam)
3	96,73	95,79	97,63	9,12
5	96,80	95,92	97,64	9,15
7	96,69	95,84	97,51	9,14

Nilai gini *impurity* index yang diperlukan untuk mencapai akurasi, sensitivitas, dan *specificity* terbaik dicapai dengan nilai gini *impurity* index 0,1 dan asumsi jumlah *subcentroid* 6. Hasil ini masih lebih baik daripada hasil TANN. Akurasi mengalami peningkatan sebesar 1,6%, sensitivitas sebesar 2,22% dan spesisitas sebesar 1,04%. Waktu pengolahan juga mengalami penurunan dari

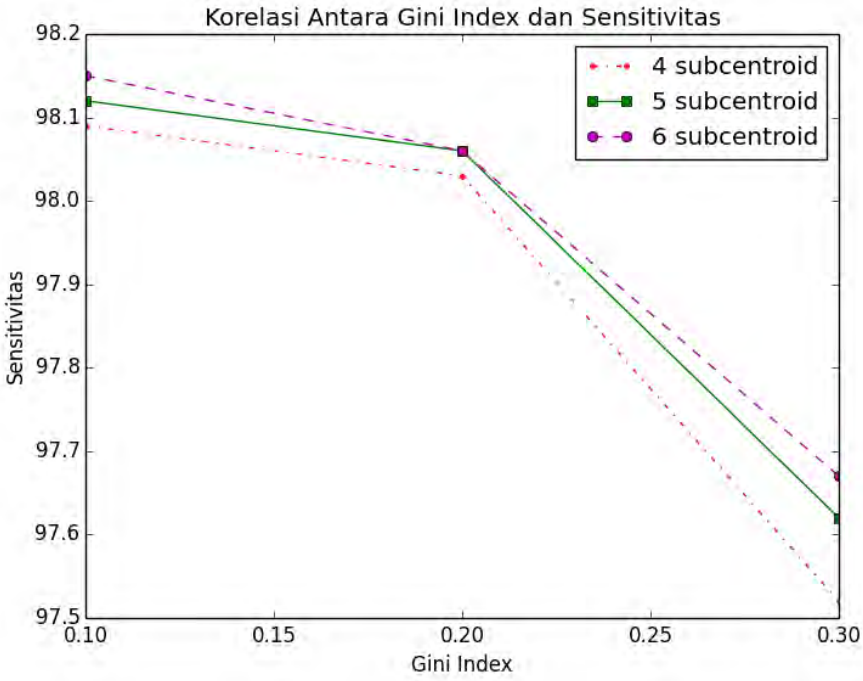
9 jam menjadi 2 jam. Waktu pengolahan metode yang diajukan tidak mengalami peningkatan sebesar TANN dimana metode ini hanya meningkat 6 kali lipat.

Tabel 4.20 Nilai Akurasi Metode yang Diajukan dengan Gini *Index* pada Kyoto2006 dalam Persen

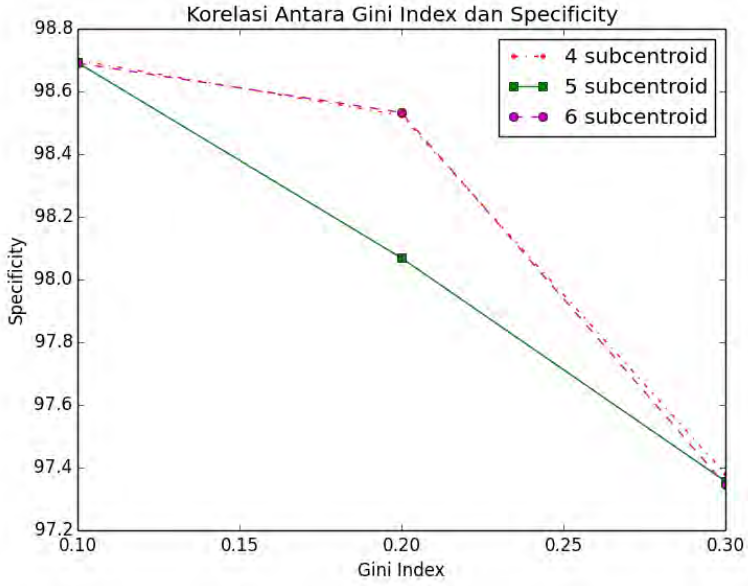
Asumsi *subcentroid*	Nilai Gini *Index*		
	0,1	0,2	0,3
4	98,40	98,28	97,45
5	98,41	98,29	97,48
6	98,42	98,30	97,50



Gambar 4.12 Korelasi antara Gini *Impurity Index* dan Akurasi pada Kyoto2006



Gambar 4.13 Korelasi antara Gini *Impurity Index* dan Sensitivitas pada Kyoto2006



Gambar 4.14 Korelasi antara Gini *Impurity Index* dan *Specificity* pada Kyoto2006

Tabel 4.21 Nilai Sensitivitas Metode yang Diajukan dengan Gini *Index* pada Kyoto2006 dalam Persen

Asumsi *subcentroid*	Nilai Gini *Index*		
	0,1	0,2	0,3
4	98,09	98,03	97,52
5	98,12	98,06	97,62
6	98,15	98,06	97,67

Tabel 4.22 Nilai *Specificity* Metode yang Diajukan dengan Gini *Index* pada Kyoto2006 dalam Persen

Asumsi *subcentroid*	Nilai Gini *Index*		
	0,1	0,2	0,3
4	98,69	98,52	97,38
5	98,69	98,06	97,35
6	98,69	98,53	97,34

Tabel 4.23 Jumlah Cluster Rata-rata dari Metode yang Diajukan dengan Gini *Impurity Index* pada Kyoto2006

Asumsi *subcentroid*	Nilai Gini *Index*		
	0,1	0,2	0,3
4	58,4	42,1	24,8
5	58,5	42	24,8
6	58,4	41,9	24,8

4.9 Hasil Metode yang Diajukan dengan *Entropy* pada Kyoto2006

Nilai akurasi, *specificity*, sensitivitas, jumlah *cluster* yang dihasilkan dan waktu pengolahan metode yang diajukan dengan *entropy* dapat dilihat pada Tabel 4.25 untuk akurasi, Tabel 4.26 untuk sensitivitas, Tabel 4.27 untuk *specificity*, Tabel 4.28 untuk jumlah *cluster*, dan Tabel 4.29 untuk waktu pengolahan. Korelasi nilai gini *impurity* index dan akurasi, sensitivitas, dan *specificity* dapat dilihat pada Gambar 4.15 untuk akurasi, Gambar 4.16 untuk sensitivitas dan Gambar 4.17 untuk *specificity*. Secara umum dampak perubahan parameter terhadap nilai akurasi, sensitivitas, dan *specificity* sama dengan gini *impurity* index. Nilai akurasi, spesifitas, dan sensitivitas terbaik dicapai dengan nilai *entropy* 0,1 dan asumsi *subcentroid* 6. Nilai terbaik dari penggunaan *entropy* lebih baik daripada penggunaan gini *impurity* index. Akurasi pada penggunaan *entropy* meningkat sebesar 0,51%, sensitivitas meningkat sebesar 0,49%, dan *specificity*

meningkat sebesar 0,53%. Secara umum waktu pengolahan mengalami penurunan dibandingkan dengan penggunaan gini *impurity* index

Tabel 4.24 Waktu Pengolahan Metode yang Diajukan dengan Gini *Impurity Index* pada Kyoto2006 dalam jam

Asumsi *subcentroid*	Nilai Gini *Index*		
	0,1	0,2	0,3
4	1,97	2,03	2,08
5	1,95	2,03	2,10
6	2,00	2,04	2,06

Tabel 4.25 Nilai Akurasi Metode yang Diajukan dengan *Entropy* pada Kyoto2006 dalam Persen

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	98,93	98,65	98,38
5	98,93	98,60	98,39
6	98,94	98,63	98,40

Tabel 4.26 Nilai Sensitivitas Metode yang Diajukan dengan *Entropy* pada Kyoto2006 dalam Persen

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	98,64	98,49	98,05
5	98,63	98,49	98,08
6	98,65	98,48	98,10

Tabel 4.27 Nilai *Specificity* Metode yang Diajukan dengan *Entropy* pada Kyoto2006 dalam Persen

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	99,21	98,79	98,69
5	99,21	98,70	98,69
6	99,22	98,70	98,69

Tabel 4.28 Jumlah Cluster Rata-rata Metode yang Diajukan dengan *Entropy* pada Kyoto2006

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	199,7	68,5	58,5
5	197,3	69,5	58,5
6	**199,4**	68,4	58,5

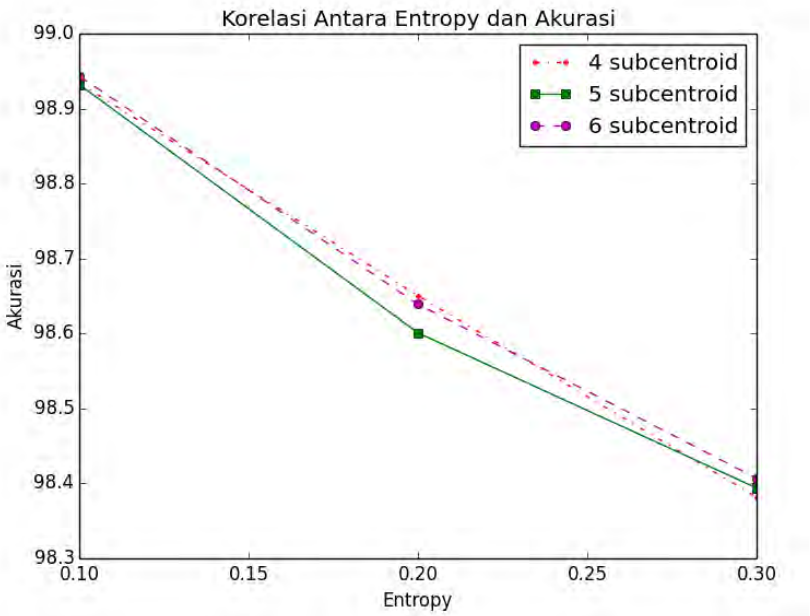
Tabel 4.29 Waktu Pengolahan Metode yang Diajukan dengan *Entropy* pada Kyoto2006 dalam Jam

Asumsi *subcentroid*	Nilai *Entropy*		
	0,1	0,2	0,3
4	1,525	1,75	1,72
5	1,58	1,82	1,86
6	1,58	1,84	1,83

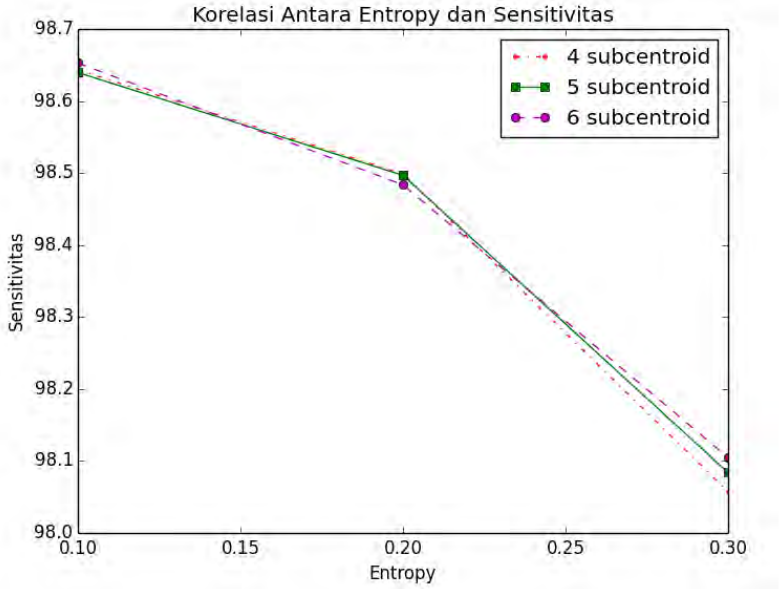
4.10 Hasil Metode yang Diajukan dengan Indeks Gabungan pada Kyoto2006

Nilai akurasi, *specificity*, sensitivitas, jumlah *cluster* yang dihasilkan dan waktu pengolahan metode yang diajukan dengan indeks gabungan dapat dilihat pada Tabel 4.30 untuk akurasi, Tabel 4.31 untuk sensitivitas, Tabel 4.32 untuk *specificity*, Tabel 4.33 untuk jumlah *cluster*, dan Tabel 4.34 untuk waktu pengolahan. Korelasi nilai gini *impurity* index dan akurasi, sensitivitas, dan *specificity* dapat dilihat pada Gambar 4.18 untuk akurasi, Gambar 4.19 untuk sensitivitas dan Gambar 4.20 untuk *specificity*. Ketiga kriteria penilaian yang dihasilkan oleh penggunaan indeks gabungan tidak terlihat perbedaan signifikan dibandingkan penggunaan *entropy*. Nilai akurasi terbesar dari indeks gabungan lebih rendah 0,03% daripada *entropy*, sensitivitas meningkat 0,03% daripada *entropy* dan *specificity* lebih rendah 0,09%. Jumlah *cluster* yang dihasilkan untuk nilai indeks gabungan dan *entropy* pada nilai 0,2 dan 0,3 tidak menunjukkan perbedaan signifikan. Namun untuk nilai 0,1 terlihat penggunaan *entropy* menghasilkan jumlah *cluster* lebih banyak daripada indeks gabungan. Waktu pengolahan indeks gabungan relatif sama dibandingkan dengan penggunaan

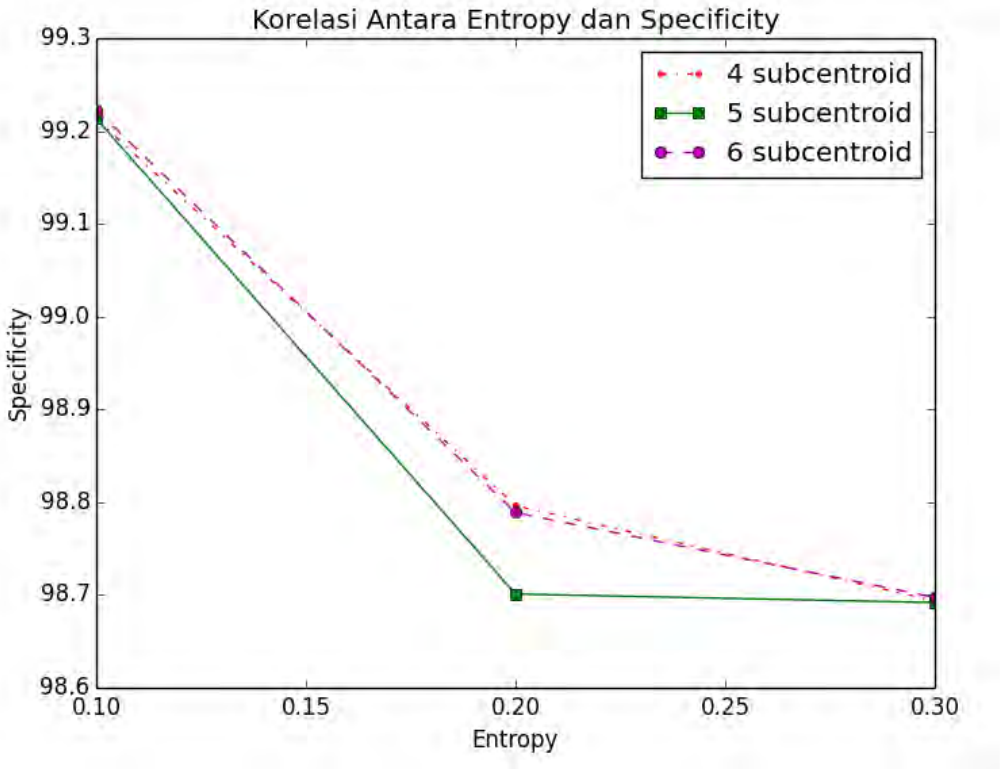
entropy. Rata-rata penurunan waktu pengolahan indeks gabungan sebesar 0,12 jam dibandingkan dengan *entropy*.



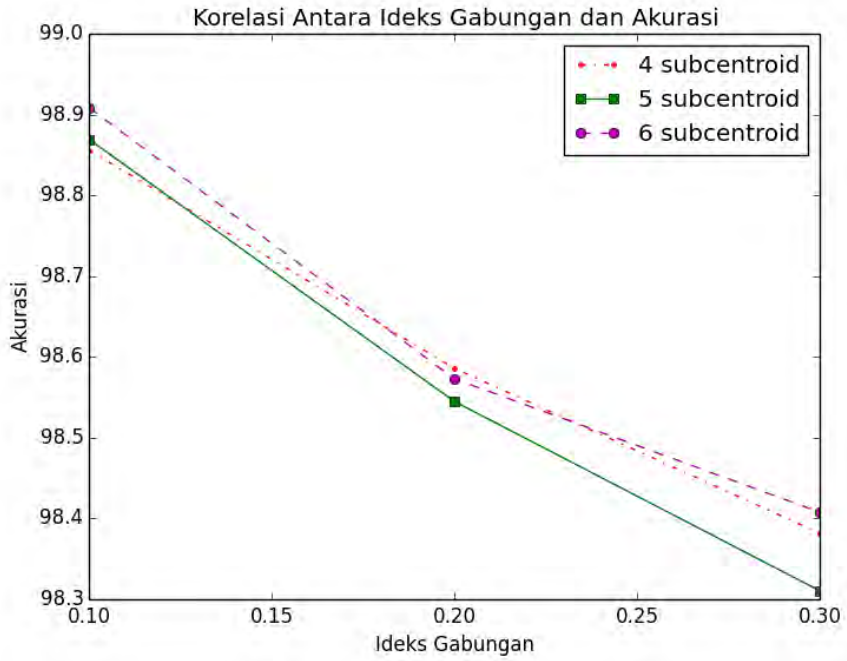
Gambar 4.15 Korelasi antara *Entropy* dan Akurasi pada Kyoto2006



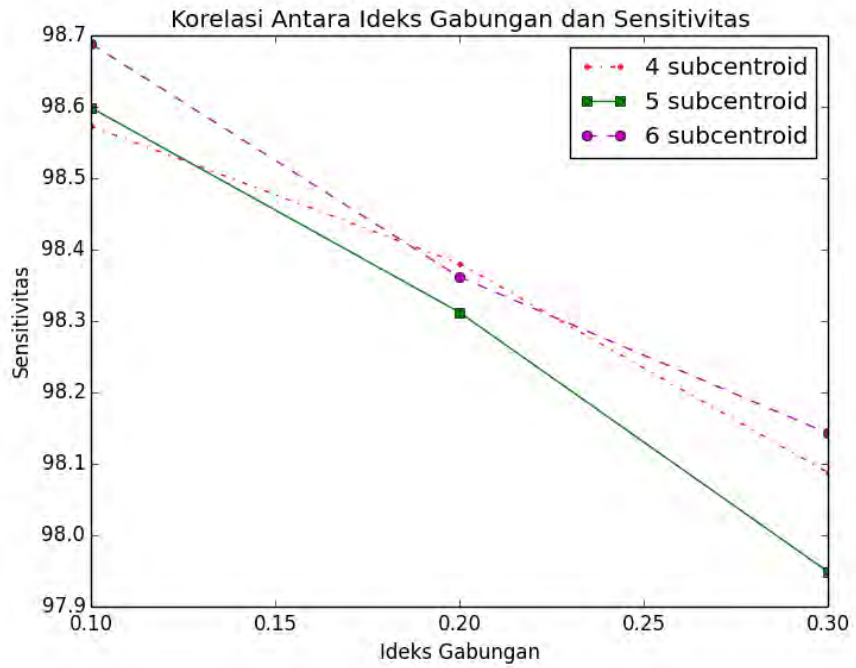
Gambar 4.16 Korelasi antara *Entropy* dan Sensitivitas pada Kyoto2006



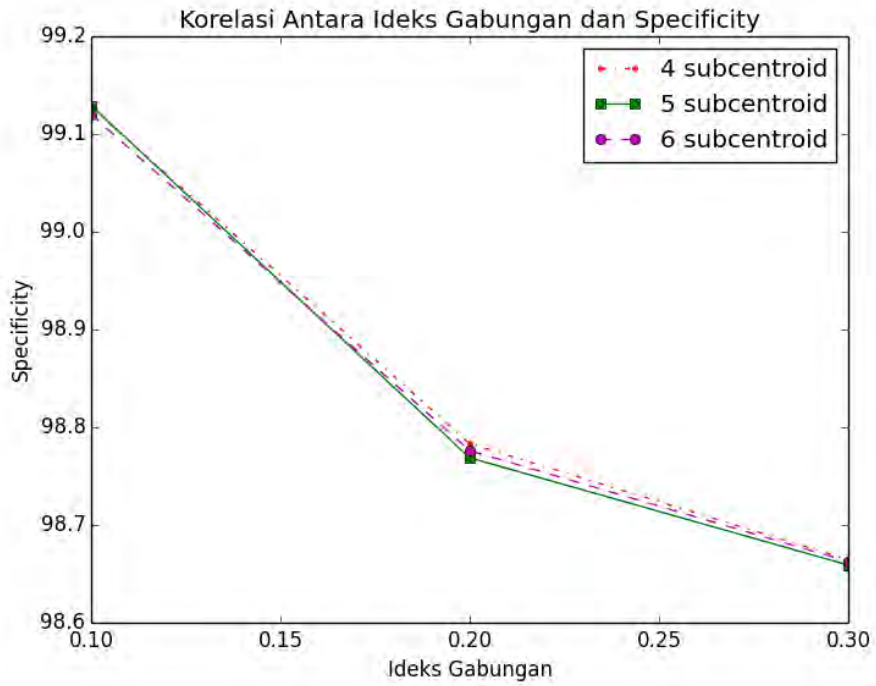
Gambar 4.17 . Korelasi antara *Entropy* dan *Specificity* pada Kyoto2006



Gambar 4.18 Korelasi antara Indeks Gabungan dan Akurasi pada Kyoto2006



Gambar 4.19 Korelasi antara Indeks Gabungan dan Sensitivitas pada Kyoto2006



Gambar 4.20 Korelasi antara Indeks Gabungan dan *Specificity* pada Kyoto2006

Tabel 4.30 Nilai Akurasi Metode yang Diajukan dengan Indeks Gabungan pada Kyoto2006 dalam Persen

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	98,85	98,58	98,38
5	98,86	98,54	98,31
6	98,90	98,57	98,40

Tabel 4.31 Nilai Sensitivitas Metode yang Diajukan dengan Indeks Gabungan pada Kyoto2006 dalam Persen

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	98,57	98,38	98,08
5	98,59	98,31	97,94
6	98,68	98,36	98,14

Tabel 4.32 Nilai *Specificity* Metode yang Diajukan dengan Indeks Gabungan pada Kyoto2006 dalam Persen

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	99,12	98,78	98,66
5	99,12	98,76	98,65
6	99,11	98,77	98,66

Tabel 4.33 Jumlah Cluster Rata-rata Metode yang Diajukan dengan Indeks Gabungan pada Kyoto2006

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	152,7	65	57,7
5	153	64,7	57,7
6	152,9	65,2	57,7

Tabel 4.34 Waktu Pengolahan Metode yang Diajukan dengan Indeks Gabungan pada Kyoto2006 dalam Jam

Asumsi *subcentroid*	Nilai Indeks Gabungan		
	0,1	0,2	0,3
4	1,59	1,72	1,73
5	1,79	1,96	1,978
6	1,80	2,03	2,00

Berdasarkan penggunaan ketida indeks diatas (gini, *entropy*, dan gabungan), nilai akurasi terbaik didapatkan pada penggunaan *entropy* dengan nilai 0,1 dan asumsi *subcentroid* 4. Kedua parameter ini menghasilkan sekitar 200 *cluster*. Jumlah *cluster* ini akan dipakai pada eksperimen berikutnya.

4.11 Hasil Metode yang Diajukan dengan Menggunakan K-*means* Pada Kyoto2006

Nilai akurasi, spesifitas, sensitivitas, jumlah *cluster* dan lama pengolahan dari metode yang diajukan dengan K-*means* biasa dapat dilihat pada Tabel 4.35. Komparasi hasil metode ini dengan TANN dan penggunaan *bisecting* K-*means* dapat dilihat pada Gambar 4.21. Berdasarkan Tabel tersebut dapat dilihat performa metode yang diajukan dengan 5 *cluster* masih lebih baik daripada TANN dengan waktu pengolahan yang sedikit lebih cepat. Salah satu faktor penyebab turunnya waktu pengolahan adalah penurunan *search space* pada fase klasifikasi. Penggunaan K-*means* dengan jumlah *cluster* yang sama pada metode yang diusulkan menyebabkan penurunan akurasi, sensitivitas dan *specificity*. Hal yang sama juga terjadi pada dataset NSL-KDD 20%. Perbedaan hasil ini dengan NSL-KDD adalah waktu pengolahan cenderung naik dari 1,806 menjadi 2,01. Kemungkinan hal ini terjadi karena perbedaan jumlah data yang diolah. Pada dataset dengan jumlah data kecil penggunaan K-*means* secara langsung dapat mengolah data lebih cepat daripada *bisecting* K-*means*, sedangkan pada dataset dengan jumlah data besar penggunaan *bisecting* K-*means* dapat mengolah data lebih cepat daripada K-*means* secara langsung.

Tabel 4.35 Hasil Metode yang Diajukan dengan K-*means* Menggantikan Bisecting K-*means* pada Kyoto2006

Jumlah Cluster	Akurasi	Sensitivitas	*Specificity*	Waktu pengolahan
5	97,70	97,49	97,91	8,84
200	98,39	97,83	98,93	2,01

4.12 Hasil Metode yang Diajukan Tanpa Logaritma pada Kyoto2006

Hasil eksperimen metode yang diajukan tanpa logaritma antara data dengan *subcentroid* dapat dilihat pada Tabel 4.36. Indeks *impurity* yang dipakai adalah *entropy* dengan nilai 0.1 dan asumsi *subcentroid* yang dipakai adalah 6. Hal yang berbeda dengan dataset NSL-KDD terjadi pada dataset ini. Terjadi peningkatan akurasi, kenaikan sensitivitas dan penurunan sensitivitas.

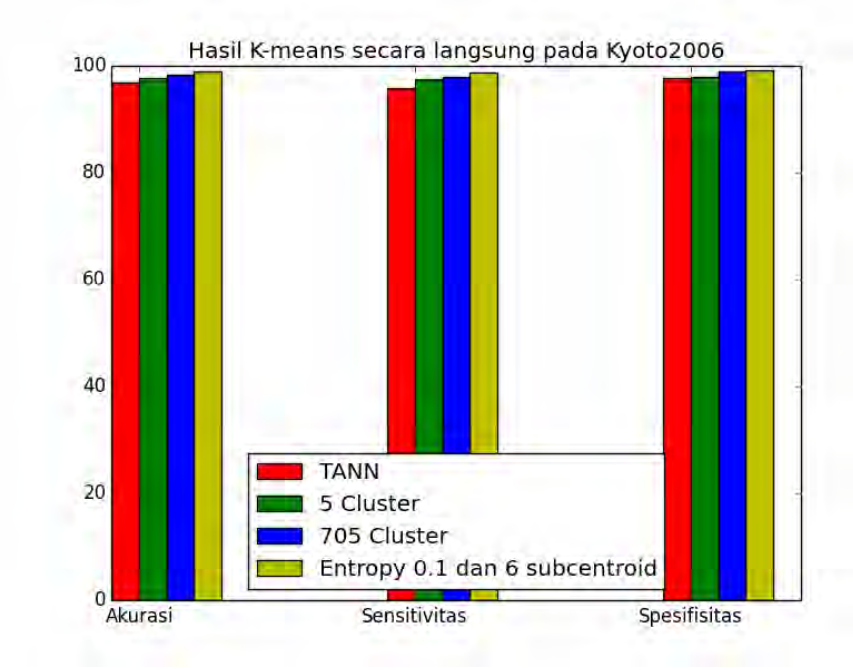
Tabel 4.36 Hasil Metode yang Diajukan tanpa Menggunakan Logaritma pada Kyoto2006

Nilai *Entropy*	Akurasi	Sensitivitas	*Specificity*	Waktu pengolahan
0,1	98,92	98,65	99,18	2,01

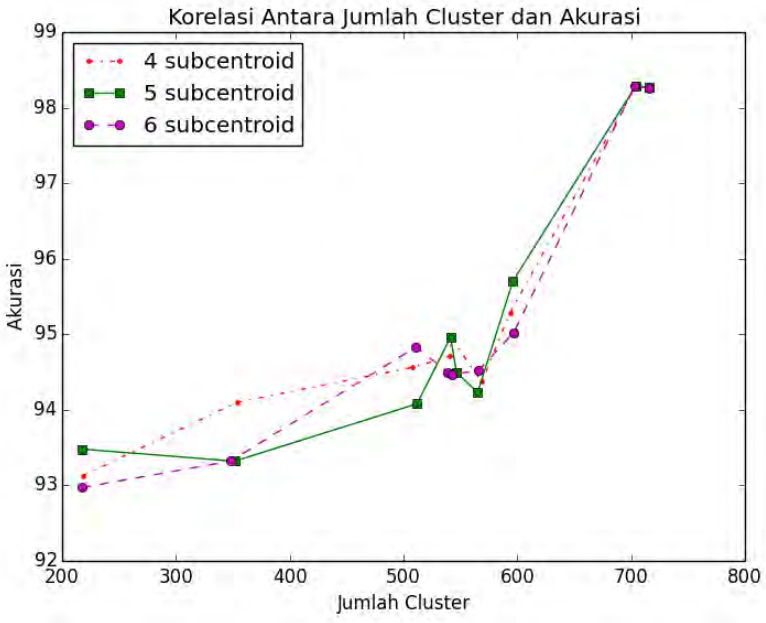
4.13 Korelasi Performa Deteksi dengan Jumlah Cluster

Korelasi jumlah *cluster* terhadap akurasi, sensitivitas, dan *specificity* dapat dilihat pada Gambar 4.22 hingga Gambar 4.27. Secara umum, penambahan jumlah *cluster* dapat berakibat pada peningkatan akurasi, sensitivitas, dan *specificity* pada kedua dataset tersebut. Perbedan yang terlihat adalah kecepatan pertumbuhan akurasi, sensitivitas, dan *specificity* dibandingkan jumlah *cluster*. Perumbuhan akurasi, sensitivitas, dan *specificity* pada Kyoto2006 lebih cepat dibandingkan dengan NSL-KDD. Hal ini disebabkan dataset Kyoto2006 terlihat lebih terpartisi secara alami daripada NSL-KDD. Hal ini ditunjukkan dengan lebih sedikitnya jumlah *cluster* yang diperlukan untuk mencapai tingkat *impurity* yang sama. Salah satu faktor hal ini adalah dataset Kyoto2006 yang dipakai berasal dari

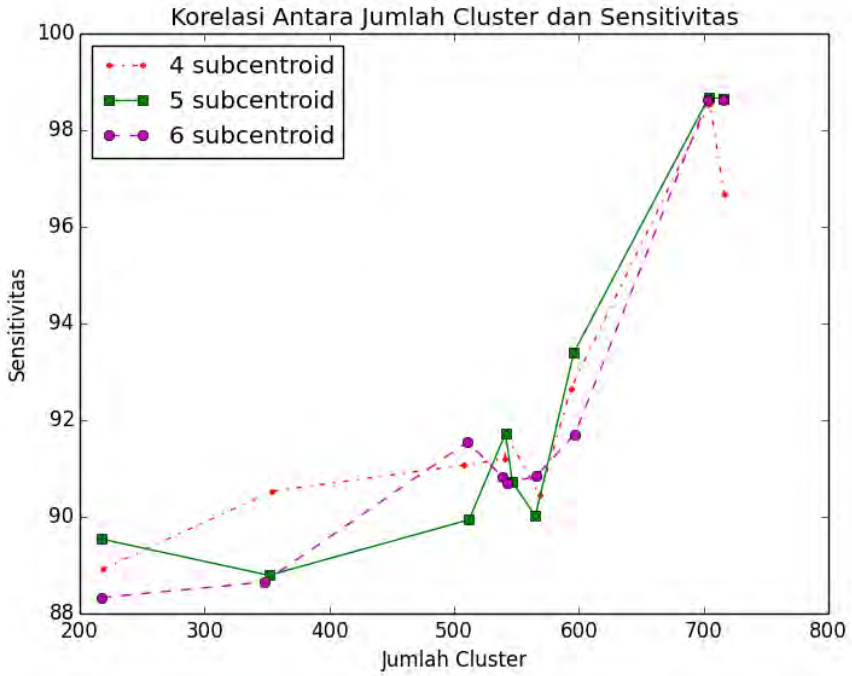
honeypot sedangkan dataset NSL-KDD yang dipakai sudah dipilih data yang relatif sulit untuk diklasifikasi.



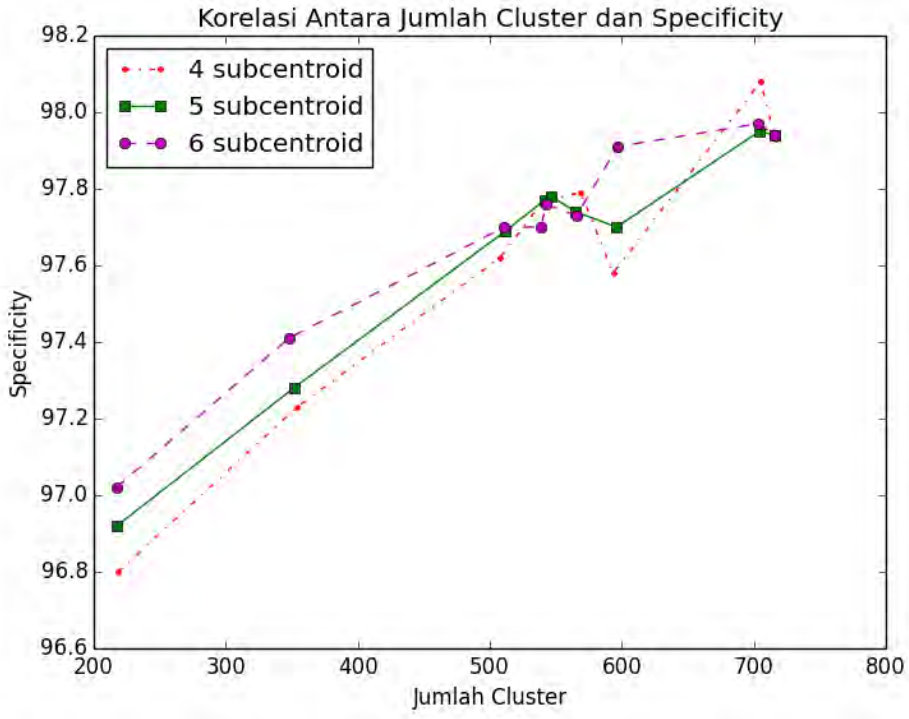
Gambar 4.21 Perbandingan K-*means* secara Langsung dengan Metode lain



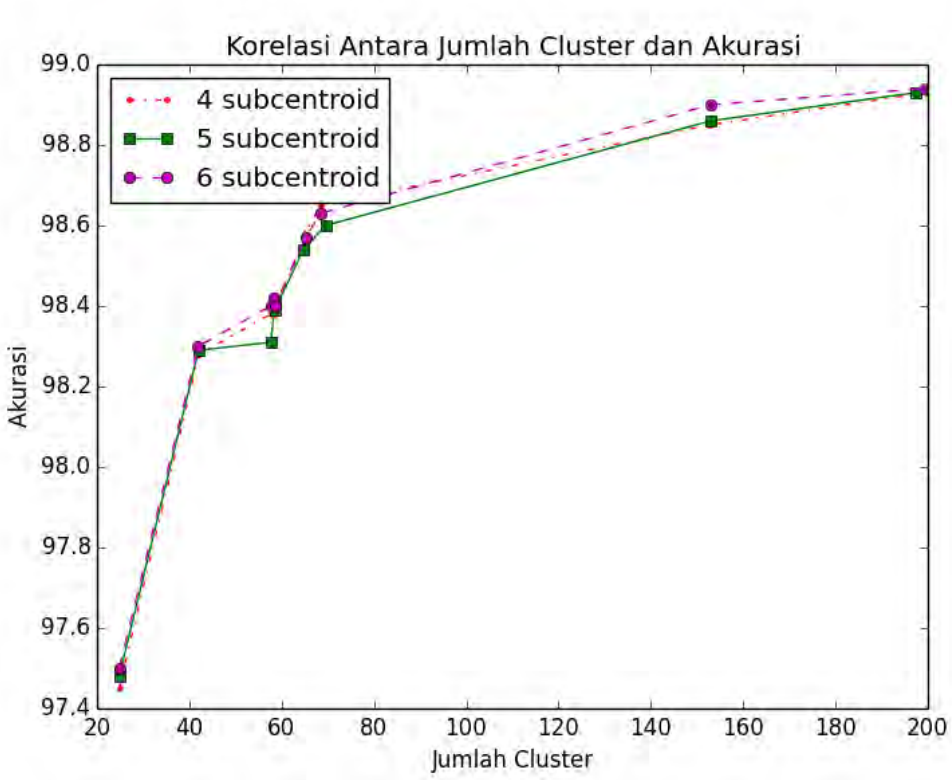
Gambar 4.22 Korelasi antara Jumlah Cluster dan Akurasi pada NSL-KDD 20%



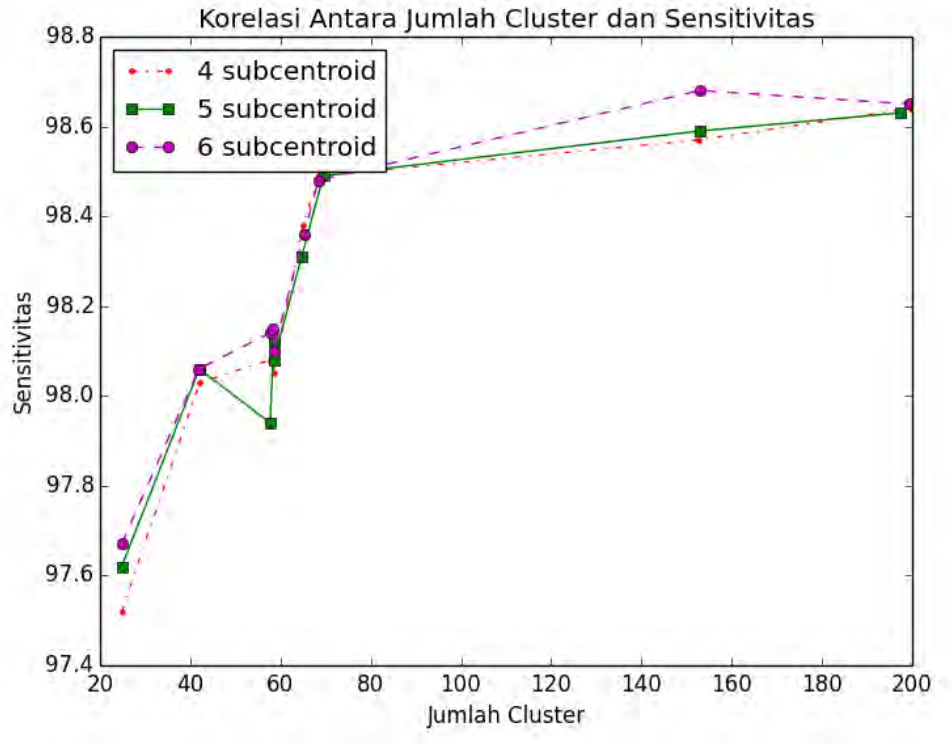
Gambar 4.23 Korelasi antara Jumlah Cluster dan Sensitivitas pada NSL-KDD 20%



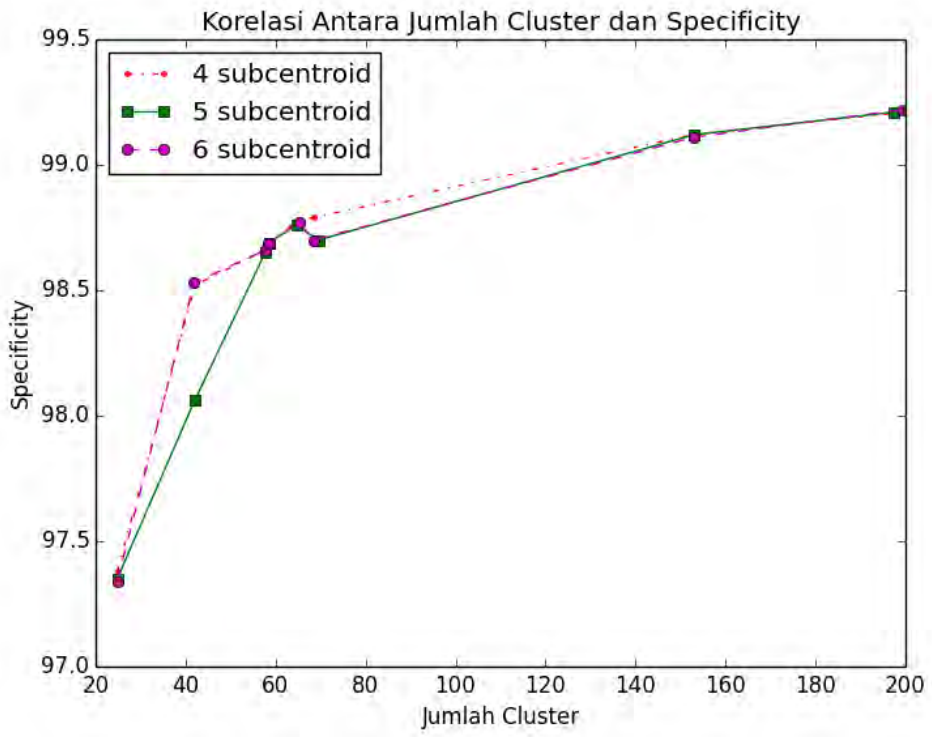
Gambar 4.24 Korelasi antara Jumlah Cluster dengan *Specificity* pada NSL-KDD 20%



Gambar 4.25 Korelasi antara Jumlah Cluster dan Akurasi pada Kyoto2006



Gambar 4.26 Korelasi antara Jumlah Cluster dan Sensitivitas pada Kyoto206



Gambar 4.27 Korelasi antara Jumlah Cluster dan *Specificity* pada Kyoto2006

4.14 Akurasi Klasifikasi pada NSL-KDD

Confusion Matrix untuk TANN dapat dilihat pada Tabel 4.37 hingga 4.39. Dapat dilihat pada Tabel-Tabel tersebut serangan yang paling mudah diklasifikasi secara tepat adalah serangan DoS dan serangan yang paling sulit diklasifikasi adalah serangan U2R (*User to Root*), (diberi warna merah). Dari seluruh tabel tersebut dapat dilihat serangan U2R kebanyakan diklasifikasi sebagai aktifitas normal. Hal ini wajar mengingat sifat dari serangan U2R sendiri yang memanfaatkan hak akses yang sudah ada untuk mencapai pengguna *root*/administrator. Kurangnya data serangan juga bisa jadi menjadi alasan sulitnya serangan ini dideteksi apalagi jika data *training* untuk serangan ini tidak membentuk suatu *cluster* dan tersebar pada *feature space*.

Confusion Matrix untuk metode yang diusulkan pada Tabel 4.40 hingga Tabel 4.66. Pada tabel tersebut dapat dilihat tingkat deteksi untuk serangan U2R meningkat dibandingkan dengan TANN meskipun tidak terdeteksi sebagai U2R.

Tingkat ketepatan klasifikasi deteksi untuk kelas serangan yang lain juga lebih baik daripada TANN.

Satu-satunya eksperimen yang berhasil menklasifikasi serangan U2R secara tepat hanya metode dengan K-*means* langsung dengan 705 cluster. Hasil deteksi pada eksperimen ini dapat dilihat pada Tabel 4.67. Dari tabel tersebut dapat dilihat terdapat 2 serangan U2R yang terklasifikasi secara tepat. Salah satu hal yang menyebabkan hal ini adalah data serangan U2R tersebut berada dalam satu *cluster* dan saling berdekatan pada *feature space.* Meski penggunaan K-*means* secara langsung dapat memberikan tingkat deteksi U2R lebih baik daripada *bisecting* K-*means,* tingkat deteksi kelas lain seperti Probe, r2l dan normal mengalami penurunan.

Tabel 4.37 *Confusion* Matrix TANN dengan K=3

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8023	533	0	7	379
Probe	985	1003	0	9	152
U2R	1	0	0	0	10
R2L	6	1	0	127	63
Normal	818	170	1	51	12310

Tabel 4.38 *Confusion* Matrix TANN dengan K=5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8228	381	0	6	286
Probe	1007	930	0	8	132
U2R	1	0	0	0	10
R2L	6	1	0	127	59
Normal	895	138	0	46	12265

Tabel 4.39 *Confusion* Matrix TANN dengan K=7

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8399	354	0	4	252
Probe	1156	909	0	6	138
U2R	1	0	0	0	10
R2L	5	3	0	126	62

Normal	931	126	0	44	12267

Tabel 4.40 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,1 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8823	350	0	0	61
Probe	302	1907	0	1	79
U2R	0	1	0	2	8
R2L	0	0	0	189	20
Normal	60	134	1	61	13192

Tabel 4.41 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,1 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8838	345	0	0	51
Probe	291	1921	0	1	76
U2R	0	1	0	2	8
R2L	0	0	0	190	19
Normal	62	150	2	61	13174

Tabel 4.42 41 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,1 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8808	369	0	0	56
Probe	291	1919	0	1	78
U2R	0	1	0	2	8
R2L	0	0	0	190	19
Normal	59	151	1	61	13177

Tabel 4.43 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,2 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8635	32	0	0	564
Probe	259	1531	0	0	499

U2R	0	1	0	0	10
R2L	0	0	0	160	49
Normal	77	119	35	64	13153

Tabel 4.44 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,2 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8610	32	0	0	587
Probe	253	1530	0	0	506
U2R	0	1	0	0	10
R2L	0	0	0	162	47
Normal	70	125	35	73	13146

Tabel 4.45 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,2 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8672	31	0	0	526
Probe	256	1529	0	0	494
U2R	0	1	0	0	10
R2L	0	0	0	164	45
Normal	75	130	35	63	13145

Tabel 4.46 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,3 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8674	32	0	0	521
Probe	303	1519	0	4	463
U2R	0	1	0	0	10
R2L	0	0	0	155	54
Normal	94	124	33	69	13129

Tabel 4.47 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,3 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8588	38	0	0	605
Probe	253	1519	0	4	513
U2R	0	1	0	0	10
R2L	0	0	0	156	53
Normal	88	124	33	65	13139

Tabel 4.48 *Confusion* Matrix Metode yang Diajukan dengan Indeks Gabungan 0,3 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8732	36	0	0	464
Probe	307	1511	0	0	465
U2R	0	1	0	0	10
R2L	0	0	0	156	53
Normal	88	119	33	69	13140

Tabel 4.49 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,1 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8471	32	0	0	459
Probe	307	1523	0	0	459
U2R	0	1	0	0	10
R2L	0	0	0	164	45
Normal	81	114	33	69	13150

Tabel 4.50 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,1 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8669	29	0	0	534
Probe	265	1524	0	1	499
U2R	0	1	0	0	10

R2L	0	0	0	163	46
Normal	80	119	33	66	13151

Tabel 4.51 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,1 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8678	33	0	0	518
Probe	265	1524	0	0	500
U2R	0	1	0	0	10
R2L	0	0	0	160	49
Normal	82	117	33	75	13141

Tabel 4.52 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,2 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8605	42	0	0	579
Probe	310	1493	0	19	466
U2R	0	1	0	0	10
R2L	0	4	0	148	57
Normal	144	131	36	58	13077

Tabel 4.53 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,2 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8473	32	0	0	717
Probe	248	1496	0	17	528
U2R	0	1	0	0	10
R2L	0	3	0	145	61
Normal	139	126	34	62	13086

Tabel 4.54 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,2 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8479	37	0	0	708
Probe	218	1494	0	18	557
U2R	0	1	0	0	10
R2L	0	3	0	149	57
Normal	126	123	36	62	13101

Tabel 4.55 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,3 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8490	48	0	0	682
Probe	270	1452	0	18	544
U2R	1	0	0	0	10
R2L	0	2	0	143	64
Normal	166	154	35	72	13019

Tabel 4.56 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,3 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8510	56	0	0	656
Probe	305	1456	0	17	505
U2R	1	0	0	0	10
R2L	0	2	0	148	57
Normal	167	140	35	71	13035

Tabel 4.57 *Confusion* Matrix Metode yang Diajukan dengan Gini *Impurity Index* 0,3 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8421	90	0	0	656
Probe	222	1460	0	17	505
U2R	1	0	0	0	10
R2L	0	3	0	148	57
Normal	151	141	35	71	13035

Tabel 4.58 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,1 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8835	342	0	0	56
Probe	291	1921	0	1	76
U2R	0	1	0	2	8
R2L	0	0	0	194	15
Normal	60	152	2	64	13171

Tabel 4.59 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,1 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8817	355	0	0	61
Probe	286	1926	0	1	76
U2R	0	1	0	3	7
R2L	0	0	0	194	15
Normal	61	146	2	67	13173

Tabel 4.60 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,1 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8806	366	0	0	61
Probe	288	1923	0	1	77
U2R	0	1	0	3	7
R2L	0	0	0	194	15
Normal	60	146	2	68	13173

Tabel 4.61 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,2 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8735	91	0	0	403
Probe	247	1596	0	1	405
U2R	0	1	0	0	10

R2L	0	0	0	163	46
Normal	81	145	35	64	13124

Tabel 4.62 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,2 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8759	117	0	0	353
Probe	288	1639	0	1	360
U2R	0	1	0	0	10
R2L	0	0	0	158	10
Normal	72	138	34	65	13140

Tabel 4.63 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,2 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8688	75	0	0	464
Probe	236	1603	0	0	449
U2R	0	1	0	0	10
R2L	0	0	0	159	50
Normal	74	113	34	60	13168

Tabel 4.64 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,3 dan Asumsi *Subcentroid* 4

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8704	36	0	0	486
Probe	276	1523	0	0	490
U2R	0	1	0	0	10
R2L	0	0	0	165	44
Normal	77	121	33	67	13150

Tabel 4.65 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,3 dan Asumsi *Subcentroid* 5

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8738	38	0	0	452
Probe	304	1526	0	0	459
U2R	0	1	0	0	10
R2L	0	0	0	160	49
Normal	80	117	33	69	13150

Tabel 4.66 *Confusion* Matrix Metode yang Diajukan dengan *Entropy* 0,3 dan Asumsi *Subcentroid* 6

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8660	33	0	0	535
Probe	261	1527	0	0	501
U2R	0	1	0	0	10
R2L	0	0	0	163	46
Normal	78	121	33	69	13148

Tabel 4.67 *Confusion* Matrix Metode yang diajukan dengan K-*means* dan 705 *cluster*

Aktual	Deteksi				
	Dos	Probe	U2R	R2L	Normal
Dos	8933	191	0	0	95
Probe	305	1799	0	0	159
U2R	0	0	2	0	9
R2L	0	1	0	0	23
Normal	120	193	3	3	13091

(halaman ini sengaja dikosongkan)

BAB 5
KESIMPULAN

Berdasarkan ekspeimen kasimpulan yang dapat diambil antara lain:

a. Pembangkitan fitur dapat dilakukan dengan menggunakan jarak ke *cnetroid* dan jarak ke *subcentroid*. Penggunaan jarak logaritma tidak terlihat memberikan perbedaan signifikan pada kualitas fitur yang dihasilkan. *Centroid* dan *subcentroid* dapat diperoleh dengan cara *bisecting* k-*means* maupun k-*means* seperti biasa, meskipun penggunaan *bisecting* K-*means* menunjukkan sedikit peningkatan dibandingkan penggunaan K-*means* secara langsung meski ukuran cluster yang dipakai rata-rata sama.
b. Faktor yang mempengaruhi homogenitas *cluster* antara lain distribusi data dalam dataset. Tingkat homogenitas ini dapat diukur dengan gini *impurity* index maupun *entropy*.
c. Penggunaan jarak ke *subcentroid* dapat meningkatkan akurasi dan mendukung pencarian lokal pada fase klasifikasi meski memakan waktu lebih lama pada fase *clustering* maupun transformasi.
d. Homogenitas *cluster* berpengaruh pada skema transformasi yang diajukan. Skema transformasi yang diajukan menunjukkan performa yang lebih baik jika proses *clustering* memperhatikan faktor homogenitas *cluster*

Saran untuk penelitian kedepan:

a. Penentuan jumlah *subcentroid* masih menggunakan cara sama rata. Hal ini menyisakan ruang untuk penelitian bagaimana mencari jumlah *subcentroid* yang paling baik baik.
b. Metode yang diajukan masih memerlukan label pada proses *training.* Proses melabeli data umumnya memerlukan waktu dan tenaga, dan dalam beberapa kasus label data tidak tersedia. Salah satu pengembangan metode ini yang disarankan adalah penggunaan *semi-supervised learning* atau *unsupervised learning* dimana data *training*

hanya sebagian yang memiliki label atau tidak memiliki label sama sekali.

DAFTAR PUSTAKA

Ahmad, T. dan Jiankun, H, (2010), "Generating cancelable biometric templates using a projection line",*11th International Conference on Control, Automation, Robotics and Vision*,IEEE, Singapura.

Ahmad, T. Hu, J. dan Han, S. (2009), "An Efficient Mobile Voting System Security Scheme based on Elliptic Curve Cryptography",*Third International Conference on Network and System Security*.IEEE, Gold Coast, hal. 474-479.

Denning, D. "An Intrusion-Detection Model",*IEEE Transactionson Software Engineering*,vol 13, no. 2, hal. 222-232.

Garcia-Teodoro, P. Diaz-Verdejo,J. Marcia-Fernandez G. dan Vazquez E. (2009), "Anomaly-based network intrusion detection: technique, system, and challenges",*Journal of Computers and Security*, 2009. hal 18-28.

Guo, C. Zhou, Y. Ping Y. Zhang, Z. Guole Liu, dan Yang, Y. (2014), "A distance sum-based hybrid method for intrusion detection",*Applied Inteligence*,vol. 40, no. 1,hal. 178-188.

Han, E-H.dan Karypis, G. (2000), "Centroid-based Document Classification Algorithms: Analysis & Experimental Results", *Proceedings of the European Conferences on Principles of Data Mining and Knowledge Discovery*, hal. 424-431.

Holil, M. dan Ahmad, T. (2015),"Secret data hiding by optimizing general smoothness difference expansion-based method.",*Journal of Theoretical and Applied Information Technology*, vol. 72, no. 2, hal. 155-163.

Lin, W.C. Ke ,S.W. dan Tsai, C.F. (2015), "CANN: An intrusion detection system based on combining *cluster* centers and nearest neighbors", *Knowledge-Based Systems*, vol 78, hal. 13-21.

McHugh, J. (2000),"Testing Intrusion Detection Systems: A Critique of the 1998 and 1999 DARPA Intrusion Detection System Evaluations as Performed by Lincoln Laboratory", *ACM Transactions on Information and System Security*,vol. 3, no. 4,hal. 262-294.

Sommer, R. dan Paxon, V. (2010),"Outside the closed world: on using machine learning for network intrusion detection", *IEEE Symposium on Security and Privacy*, IEEE, Oakland, hal. 305-316.

Song, J. Takakura, H. Okabe, Y. Eto, M. Inoue, D. Nakao, K. (2011), "Statistical analysis of honeypot data and building of Kyoto2006+ dataset for NIDS evaluation", *Workshop on Development of Large Scale Security-related Data Collection and Analysis Initiatives*, ACM, Salsburg, hal 29-36.

Steinbach, M. Karypis, G. Kumar, V. (2000), "A Comparison of Document Clustering Technique", *KDD workshop on text mining.*

Tavallaee, M. Bagheri, E. Liu, W. dan Ghorbani, A. (2009) "A Detailed Analysis of the KDD CUP 99 Data Set", *IEEE Symposium on Computational Intelligence for Security and Defense Applications*.IEEE, Ottawa, hal. 1-6.

Tsai, CF. dan Lin, CY. (2010), "A triangle area based nearest neighbors approach to intrusion detection.", *Pattern Recognition*, vol. 43, hal. 222–229.

Witten, I.H. dan Frank, E. (2005), *Data Mining: Practical Machine Learning Tools and Technique*,Morgan Kaufmann.,San Francisco.

BIOGRAFI PENULIS

Penulis, Kharisma Muchammad, lahir di Surabaya 3 Maret 1990. Putra pertama dari pasangan Hadi Surono dan Endang Murnia. Penulis menempuh pendidikan SD di SDN Wedoro I (1996-2002), SMP di SMP I Waru (2002-2005), SMA di SMA 15 Surabaya (2005-2008), dan S1 di Jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember (ITS) (2008-2013).

Dalam menempuh pendidikan S1, penulis mengambil bidang minat Rekayasa Perangkat Lunak, dan pada pendidikan S2 penulis mengambil bidang minat Komputasi Bebasis Jaringan. Penulis dapat dihubungi lewat email kharisma_muchammad@yahoo.co.id.